As we saw in the previous chapter’s comparison of recognition and recall, the human brain is good at some things and not so good at others. In this chapter, we compare several additional functions of the brain to show which functions it is good and bad at, and to see how to design computer systems accordingly. But first, a bit more about the brain and the mind.

**WE HAVE THREE BRAINS**

We really have three brains, or if you prefer, a brain with three main parts, each of which affects different aspects of our thought and behavior (Weinschenk, 2009):

- **The old brain**: This is mainly the brain stem, where the spinal cord enters the base of the brain. It has been around since the first fish evolved. (Insects and mollusks, which appeared before fish, don’t have brains in the usual sense of the word.) The old brain classifies everything into three categories: edible, dangerous, or sexy. It also regulates the body’s automatic functions, such as digestion, breathing, and reflexive movement. Reptiles, amphibians, and most fish have only the old brain.

- **The midbrain**: This part of the brain is “middle” in two senses: (a) physically, because it is located above the old brain and beneath the cortex, and (b) evolutionarily, because it evolved after the old brain and before the new brain. The midbrain controls emotions; it reacts to things with joy, sadness, fear, aggressiveness, apprehensiveness, anger, etc. Birds\(^1\) and lower mammals have only an old brain and a midbrain.

\(^1\)Corvids (ravens, crows, and magpies) and some types of parrots (e.g., New Zealand kia) have no cortex, but they do have large brains compared to other birds. They often exhibit intelligence rivaling elephants, porpoises, and monkeys. In these birds, other parts of their brains apparently serve functions that the cortex serves in mammals.
CHAPTER 10  Problem Solving and Calculation are Hard

- **The new brain**: This part of the brain mainly consists of the cerebral cortex. It controls intentional, purposeful, conscious activity, including planning. Most mammals have a cortex in addition to their old brain and midbrain, but only a few highly evolved mammals—elephants; porpoises, dolphins, and whales; and monkeys, apes, and humans—have a sizable one.

The human mind is not fully rational and conscious—some experts claim that it isn’t even *mostly* rational and conscious. Our thoughts and behavior are affected at least as much by the midbrain and old brain as they are by the new brain. When we perceive something—an object or an event—all three “brains” react and contribute to our thought and behavior. In fact, the old and midbrains tend to react faster than the new brain does, so we sometimes act based on what they tell us before our cortex reaches a decision or even knows that action is required.

**LEARNING FROM EXPERIENCE IS (USUALLY) EASY**

People are pretty good at generalizing from specific experiences and observations to extract conclusions. We generalize constantly throughout our lives.

The neural basis of behavioral learning is not as well understood as that of recognition and recall (Liang *et al.*, 2007). However, people learn from their experiences constantly, often without awareness that they are doing it. From this fact we can postulate that the human brain evolved the ability to learn quickly and easily from experience because there were evolutionary advantages to being able to do so. Thus, most people, if given the necessary experience, easily learn such lessons as these:

- Stay away from leopards
- Don’t eat bad-smelling food
- Ice cream tastes good, but it melts quickly in hot weather
- Wait a day before replying to an email that makes you mad
- Don’t open attachments from unfamiliar senders
- LinkedIn is useful, but Facebook is a waste of time (or vice versa, depending on your preference)

However, our ability to learn from experience is not perfect for several reasons. First, complex situations that involve many variables or that are subject to a wide variety of forces are difficult for people to predict, learn from, and generalize about. For example:

- Experienced stock market investors still aren’t sure what stocks to sell or buy now.
- People who have lived in Denver for years still have trouble predicting the weather there.
- Even after interacting with your sister’s boyfriend on several occasions, you may still not be sure he is a good guy.

Second, experiences from our own lives or those of relatives and friends influence our conclusions more than experiences we read or hear about. For example, we may have read and seen reports, consumer reviews, and statistics indicating
that the Toyota Prius is a great car, but if our sister or our uncle had a bad experience with one, we will probably have a negative assessment of the car. We do this because our midbrain considers family members to be like us and therefore more trustworthy than data about thousands of anonymous car buyers, even though from a rational standpoint the statistics are more reliable (Weinschenk, 2009).

Third, when people make a mistake, they don’t always learn the right lesson from it. By the time they realize they are in a bad situation, they may not remember their recent actions well enough to be able to connect their situation with the true cause or causes.

A fourth problem people have in learning from experience is that they often overgeneralize, i.e., make generalizations based on incomplete data. For example, many people assume all crows are black because all the crows they have seen are black. In fact, there are crows that are not black (see Fig. 10.1).

However, it can be argued that overgeneralizing isn’t a problem—it’s a feature. It is rare that one can see all possible examples of something. For example, a person can never see all crows, but it may still be useful in daily life (although not in scientific research) to assume that the many crows one has seen are enough evidence to conclude that all crows are black. Overgeneralization therefore seems like a necessary adaptation for life in the real world. It is primarily when we overgeneralize in extreme ways—e.g., making generalizations on the basis of one example or atypical examples—that we get ourselves into trouble.

The ability to learn from experience has a long evolutionary history. A creature does not need a cerebral cortex (new brain) to be able to do it. Both the old brain and midbrain can learn from experience. Even insects, mollusks, and worms, without even an old brain—just a few neuron clusters—can learn from experience. However, only creatures with a cortex or brain structures serving similar functions² can learn

![Figure 10.1](image.png)

**FIGURE 10.1**
The common belief that all crows are black is false. Left: African pied crow (Photograph by Thomas Schoch). Right: white (nonalbino) crow, Ohio.

²The reason for the caveat is that some birds can learn from watching other birds.
from the experiences of others. A cortex is certainly necessary to be aware that one has learned from experience, and only creatures with the largest new brains (relative to body size)—possibly only humans—can articulate what they have learned from experience.

Even though there are limits on how well we learn from direct experience and from the experience of others, the bottom line is that learning and generalizing from experience are relatively easy for the human mind.

PERFORMING LEARNED ACTIONS IS EASY

When we go somewhere we have been many times before, or do something we have done many times before, we do it almost automatically, without much conscious thought. The route, the routine, the recipe, the procedure, the action, has become semiautomatic or fully automatic. Here are some examples:

- Riding a bicycle after many years of practice
- Backing out of your driveway and driving to work for the 300th time
- Brushing your teeth as an adult
- Playing a tune that you have played hundreds of times on a musical instrument
- Using a mouse or a touch pad to move a cursor on a computer display after a few days of practice
- Entering a banking transaction into your old familiar bank account software
- Reading and then deleting a text message from your longtime mobile phone

In fact, “automatic” is how cognitive psychologists refer to routine, well-learned behavior (Schneider & Shiffrin, 1977). Researchers have determined that performing this type of action consumes few or no conscious cognitive resources, i.e., it is not subject to the limits of attention and short-term memory described in Chapter 7.

Automatic activities can even be done in parallel with other activities. Thus, you can tap your foot while humming a familiar song while beating an egg, while still leaving your mind “free” to keep an eye on your children or plan your upcoming vacation.

How does an activity become automatic? The same way you get to Carnegie Hall (as the old joke goes): practice, practice, practice.

When a person first tries to drive a car—especially a car with a stick shift—every part of the activity requires conscious attention. Am I in the right gear? Which foot do I use to press the accelerator pedal, the brake pedal, and the clutch pedal? How hard should I press on each of these pedals? How hard am I pressing on the clutch pedal now? Which way am I headed? How fast am I going? What is ahead of me, behind me, beside me? Where are the mirrors I should be checking? Is that my street coming up ahead? “Objects in mirror are closer than they appear”—what does that mean? And what is that light blinking on the dashboard?

When everything involved in driving a car is still conscious, keeping track of it all far exceeds our attention capacity—remember that it is four items, plus or minus two (see Chapter 7). People who are still learning to drive often feel overwhelmed. That is
Performing learned actions is easy

why they often practice driving in parking lots, parks, rural areas, and quiet neighborhoods, where traffic is light: to reduce the number of things they have to attend to.

After a lot of practice, all the actions involved in driving a car become automatic. They no longer compete for attention and they recede from consciousness. We may not even be fully aware of doing them. For example, which foot do you use to push the accelerator pedal? To remember, you probably had to pump your feet briefly.

Similarly, when music teachers teach students to play a musical instrument, they don’t make students monitor and control every aspect of their playing at once. That would overwhelm the students’ attention capacity. Instead, teachers focus students’ attention narrowly on one or two aspects of their playing: the correct notes, the rhythm, the tone, the articulation, or the tempo. Only after students learn to control some aspects of their playing without thinking about them do music teachers require their students to control more aspects simultaneously.

To demonstrate to yourself the difference in conscious attention required by well-learned (automatic) versus novel (controlled) tasks, try these:

• Recite the letters of the alphabet from A to M. Then recite the letters of the alphabet from M to A.

• Count down from 10 to 0—think of a rocket launch. Then count down from 21 to 1 by odd numbers.

• Drive to work, using your normal route. The next day, use a very different, unfamiliar route.

• Throw a ball with your usual ball-throwing hand. Then throw one using the opposite hand.

• Enter your phone number using a standard 12-key telephone pad. Then enter your phone number using the number keys at the top of your computer keyboard.

• Type your full name on a computer keyboard. Then cross your hands on the keyboard and type your full name again. (I was going to suggest riding a bicycle with your hands crossed, but that is actually dangerous, so I do not recommend trying it.)

Most real-world tasks have a mixture of automatic and controlled components. Driving to work along your usual route is mostly automatic, allowing you to focus on the radio news or think about your evening dinner plans. But if another vehicle near you does something unexpected or a child appears on the road ahead of you, your attention will be yanked back to the task of driving.

Similarly, if you check your email using your usual email program, the way you retrieve and view your email is well practiced and mostly automatic, and reading text is well practiced and automatic, but the content of any newly arrived email messages is new and therefore requires your conscious attention. If while on vacation you go into an Internet cafe and try to check your email using an unfamiliar computer, operating system, or email program, less of the task will be automatic, so it will require more conscious thought, take more time, and be more prone to error.
When people want to get something done—as opposed to challenging themselves mentally—they prefer to use methods that are automatic or at least semi-automatic in order to save time and mental effort, and to reduce the chance of error. If you are in a hurry to pick up your child from school, you take your tried-and-true route, even if your neighbor just told you yesterday about a faster route. Remember what the usability test subject said (previously mentioned in Chapter 8):

I'm in a hurry, so I'll do it the long way.

How can designers of interactive systems make the tasks that they support faster, easier, and less error prone? By designing them to become automatic quickly. How does one do that? Chapter 11 describes some of the ways.

PROBLEM SOLVING AND CALCULATION ARE HARD

Reptiles, amphibians, and most birds get along in their world quite well with just an old brain and a midbrain. Insects, spiders, and mollusks survive in their environments with even less. Animals without a cortex (or its equivalent, as in a few birds) can learn from experience, but it usually takes a lot of experience and they can only learn minor adjustments to their behavior. Most of their behavior is stereotyped, repetitive, and predictable once we understand the demands of their environment (Simon, 1969). That may be just fine when their environment requires only the behaviors they already have automated.

But what if the environment throws a curve ball: it requires new behavior, and requires it right now? What if a creature faces a situation it has never encountered before, and may never encounter again? In short, what if it is faced with a problem? In such cases, creatures with no cortex or its equivalent cannot cope.

Having a cerebral cortex (new brain) frees creatures from relying solely on instinctive, reactive, automatic, well-practiced behaviors. The cortex is where conscious reasoning happens (Monti, Osherson, Martinez, & Parsons, 2007). Generally speaking, the larger a creature’s cerebral cortex relative to the rest of its brain, the greater its ability to interpret and analyze situations on-the-fly, plan or find strategies and procedures to cope with those situations, execute those strategies and procedures, and monitor their progress.

Expressed in computer jargon, having a large cortex gives us the ability to devise programs for ourselves on the fly and run them in an emulated, highly monitored mode rather than a compiled or native mode. That is essentially what we are doing when we are following a cooking recipe, playing bridge, calculating income taxes, following instructions in a software manual, or figuring out why no sound is coming out of the computer when we play a video.

---

3For example, salamanders choose a jar containing four fruit flies over one with two or three fruit flies (Sohn, 2003).
THE NEW BRAIN ALSO ACTS AS A BRAKE ON IMPULSIVE BEHAVIOR

The new brain—specifically the frontal cortex—also acts to inhibit reflexive and impulsive behavior coming from the midbrain and old brain that could interfere with the execution of the new brain’s carefully worked-out plans (Sapolsky, 2002). It keeps us from jumping up and getting off of a subway car when a smelly person boards, because after all, we do have to get to work on time. It keeps us sitting quietly in our seats in classical music concerts, but lets us stand up and hoot and holler in rock concerts. It helps keep us out of fights (usually). It tries to stop us from buying that red sports car because preserving our marriage is a higher goal than having the car. And whereas the old and midbrains are tempted by the email that proposes a BUSINESS OPPORTUNITY WORTH $12.5 MILLION, the new brain stops us from clicking, saying “It’s a spammer and a scammer; you know that, don’t you?”

Although having a large new brain gives us the flexibility to deal with problems on short notice, that flexibility has a price. Learning from experience and performing well-learned actions are easy largely because they don’t require constant awareness or focused attention and because they can occur in parallel. In contrast, controlled processing—including problem solving and calculation—requires focused attention and constant conscious monitoring, and executes relatively slowly and serially (Schneider & Shiffrin, 1977). It strains the limits of our short-term memory because all the chunks of information needed to execute a given procedure compete with each other for scarce attention resources. It requires conscious mental effort, as you saw when you tried to recite the alphabet backward from M to A.

In computer jargon, the human mind has only one serial processor for emulation mode, controlled execution of processes. That processor is severely limited in its temporary storage capacity and its clock is an order of magnitude slower than that of the brain’s highly parallelized and compiled automatic processing.

Modern humans evolved from earlier hominids between 200,000 and 50,000 years ago, but numbers and numerical calculation did not exist until about 3400 BC, when people in Mesopotamia (modern-day Iraq) invented and started using a number system in commerce. By then, the human brain was more or less as it is today. Since the modern human brain evolved before numerical calculation existed, it is not optimized for calculation.
Calculation is done mainly in the brain’s controlled, monitored mode. It is a task that consumes scarce resources of attention and short-term memory, so when we try to perform calculations entirely in our heads, we have trouble. The exception is that some steps in a calculation may be memorized and therefore are automatic. For example, the overall process of multiplying $479 \times 832$ is controlled, but certain substeps of the process may be automatic if we have memorized the multiplication tables for single-digit numbers.

Problems and calculations that involve only one or two steps, or in which some steps are memorized (automatic), or that don’t involve much information, or in which all the relevant information is immediately available—and therefore need not be kept in short-term memory—are easy for most people to work out in their heads. For example:

- $9 \times 10 = ?$
- I need to move the washing machine out of the garage, but the car is in the way, and my car keys are in my pocket. What to do?
- My girlfriend has two brothers, Bob and Fred. I have met Fred, and the one here now isn’t Fred, so it must be Bob.

However, problems that exceed our short-term memory limits, or that require that certain information be retrieved from long-term memory, or in which we encounter distractions, strain our brains. For example:

- I need to move the washing machine out of the garage, but the car is in the way, and my car keys are ... hmmm ... they’re not in my pocket. Where are they? ... [Search car.] They’re not in the car. Maybe I left them in my jacket. ... Now where did I leave my jacket? [Search house; eventually find jacket in bedroom.] OK, found the keys. ... Boy is this bedroom messy—must clean it before wife gets home. ... Hmmm. Why did I need the car keys? [Return to garage, see washer.] Oh, yeah: to move the car so I can move the washing machine out of the garage. *(Higher-level goal was pushed out of short-term memory by interim subgoals.)*

- Chapter 8 gave examples of tasks in which people have to remember to complete cleanup steps after achieving their primary goal, for example, remembering to turn your car headlights OFF after arrival at your destination or to remove the last page of a document from a copier after you have the copy.

- John’s cat is not black and likes milk. Sue’s cat is not brown and doesn’t like milk. Sam’s cat is not white and doesn’t like milk. Mary’s cat is not yellow and likes milk. Someone found a cat that is yellow and likes milk. Whose cat is it?\(^4\) *(The negations create more chunks of information than most people’s short-term memory can hold at once.)*

---

\(^4\)Answers provided at the end of this chapter.
• A man built a four-sided house. All four walls faced south. A bear walked by. What color was the bear? *(Requires deduction and knowing and retrieving specific facts about the world and its wildlife.)*

• You have to measure exactly four liters of water, but you only have a three-liter bottle and a five-liter bottle. How do you do it? *(Requires mentally simulating a series of pours until the right series is found, straining short-term memory and perhaps exceeding mental simulation abilities.)*

When solving such problems, people often use external memory aids, such as writing down interim results, sketching diagrams, and manipulating models of the problem. Such tools augment our limited short-term memory and our limited ability to imagine manipulating problem elements.

Problem solving and calculation are also difficult if they require a cognitive strategy, solution method, or procedure that we don’t know and cannot devise or find. For example:

• $93.3 \times 102.1 = ?$ *(Requires arithmetic that exceeds short-term memory capacity, so must be done with a calculator or on paper. The latter requires knowing how to multiply multidigit decimal numbers on paper.)*

• A farmer has cows and chickens—30 animals total. The animals have a total of 74 legs. How many of each animal does the farmer have? *(Requires translation to two equations and then solving using algebra.)*

• A Zen master blindfolded three of his students. He told them that he would paint either a red dot or a blue dot on each one’s forehead. In fact, he painted red dots on all three foreheads. Then he said “In a minute I will remove your blindfolds. When I do, look at each other and if you see at least one red dot, raise your hand. Then guess which color your own dot is.” Then he removed the blindfolds. The three students looked at each other, then all three raised a hand. After a minute, one of the students said “My dot is red.” How did she know? *(Requires reasoning by contradiction, a specialized method taught in logic and mathematics.)*

• You play a YouTube video on your computer, but there is no sound even though you can see people speaking. Is the problem in the video, the video player, your computer, your speaker cables, or your speakers? *(Requires devising and executing a series of diagnostic tests that successively narrow the possible causes of the problem, which requires computer and electronics domain knowledge.)*

These made-up examples demonstrate that certain problems and calculations require training that many people do not have. The sidebar gives real examples of people being unable to resolve technical problems because they lack training in effective diagnosis in the technical problem domain and are not interested in learning how to do it.
SOLVING TECHNICAL PROBLEMS REQUIRES TECHNICAL INTEREST AND TRAINING

Software engineers are trained to do systematic diagnosis of problems. It is part of their job to know how to devise and execute a series of tests to eliminate possible causes of a fault until they find the cause. Engineers often design technology-based products as if the intended users were as skilled as engineers in diagnosing technical problems. However, most people who are not software engineers have not been trained in that sort of problem diagnosis, and therefore cannot do it effectively. Here are real examples of non-technical people facing problems they could not solve without help:

- Ann wanted to book a flight, but couldn’t because the airline Web site wouldn’t let her. It demanded a password but she didn’t have one. She called a computer-engineer friend, who asked several questions to learn her situation. It turned out that the Web site assumed that she was her husband, because he had previously bought tickets from that airline on that computer. The site wanted his username and password. She didn’t know his password and he was out of town. The engineer told her to log out of the Web site, then return as a new customer and create her own account.

- At a church, one of two stage monitor speakers stopped working. The Assistant Music Director assumed that the monitor had failed and said he would replace it. A musician who also is an engineer wasn’t sure the monitor was bad, so he swapped the two monitor cables at the speaker end. Now the “bad” speaker worked and the “good” one didn’t, showing that the problem was not a bad speaker. The Assistant Music Director concluded that one speaker cable was bad and said he would buy a new one. Before he did, the engineer-musician swapped the monitor cables where they connect to the monitor amplifier, to see if the problem was a faulty monitor.
amplifier output rather than the cable. The problem turned out to be a loose connection in the monitor amplifier output jack.

Even when people know they could solve a problem or perform a calculation if they put effort into it, sometimes they don’t do it because they don’t consider the potential reward worth the effort. This response is especially common when solving a problem is not required by one’s job or otherwise. Here are some real examples:

- A posting on San Francisco Freecycle Network: “Free: Epson Stylus C86. Was working fine, and then suddenly it couldn’t recognize the new full ink cartridge. Not sure if it’s the cartridge or the printer. So I bought a new printer and am giving the old one away.”

- Fred and Alice, a schoolteacher and a nurse who are married, never install or update software on their home computer. They don’t know how, and they don’t want to know. They use only the software that came with the computer. If their computer says updates are available, they ignore it. If an application—e.g., a Web browser—stops working because it is outdated, they stop using it. When necessary, they buy a new computer.

- Another couple, Ted and Sue, have a television, a videotape player, and a DVD player. Remote controls for the devices lie in a pile near the TV, unused. Ted and Sue control the devices by getting up and walking across the room. They say it’s too much trouble to learn to work the remotes and remember which one is for which device. Yet they use computers daily, for email and Web.

The people in these examples are not stupid. Many have college degrees, putting them in the top 30% of educational attainment in the United States. Some are even trained to diagnose problems in different domains, such as medicine. They just have no training or interest in solving technical problems in computers and computer-based devices.
People invented calculators and computers mainly as tools for performing calculations and solving problems that humans cannot easily solve on their own. Computers and calculators do calculation and problem solving much more easily and reliably than we do, at least when the problems are well defined.

**IMPLICATIONS FOR USER INTERFACE DESIGN**

People often intentionally challenge and entertain themselves by creating or solving puzzles that strain—or “exercise”—their minds (see Fig. 10.2). However, that fact does not imply that people will happily accept mind-straining problems foisted upon them by someone or something else. People have their own goals. They are using a computer to help them achieve a goal. They want—and need—to focus their attention on that goal. Interactive systems—and designers of them—should respect that and not distract users by imposing technical problems and goals that users don’t want.

Here are some examples of technical problems that computers and Web services impose upon their users:

- “It wants my ‘member ID.’ Is that the same as my ‘username’? It must be.”
- “Huh? It charged me the full price! It didn’t give me my discount. What now?”
- “It says that the software may be incompatible with a plug-in already on my computer. ‘May be?’ Is it or isn’t it? And if it is, which plug-in is the culprit? What should I do?”
- “I want page numbers in the chapter to start at 23 instead of 1, but I don’t see a command to do that. I’ve tried Page Setup, Document Layout, and View Header and Footer, but it isn’t there. All that’s left is this Insert Page Numbers command. But I don’t want to insert page numbers: the chapter already has page numbers. I just want to change the starting number.”
- “Hmmm. This checkbox is labeled Align icons horizontally. I wonder what happens if I uncheck it. Will my icons be aligned vertically, or will they simply not be aligned?”

**FIGURE 10.2**
We challenge ourselves by creating and solving puzzles that tax our mental abilities.
Interactive systems should minimize the amount of attention users must devote to operating them (Krug, 2005), because that pulls precious cognitive resources away from the task a user came to the computer to do. Here are some design rules:

- **Prominently indicate system status and users’ progress toward their goal.** If users can always check their status easily by direct perception, using the system will not strain their attention and short-term memory.

- **Guide users toward their goals.** Designers can do this implicitly, by making sure every choice-point provides clear information “scent” that leads users toward their goal, or explicitly, by using a wizard (multistep dialog box). Don’t just display a bunch of options that appear equally likely and expect users to know how to start and how to get to their goal, especially if they won’t perform the task very often.

- **Tell users explicitly and exactly what they need to know.** Don’t expect them to deduce information. Don’t require them to figure things out by a process of elimination.

- **Don’t make users diagnose system problems,** such as a faulty network connection. Such diagnosis requires technical training, which most users don’t have.

- **Minimize the number and complexity of settings.** Don’t expect people to optimize combinations of many interacting settings or parameters. People are really bad at that.

- **Let people use perception rather than calculation.** Some problems that might seem to require calculation can be represented graphically, allowing people to achieve their goals with quick perceptual estimates instead of calculation. A simple example: suppose you want to go to the middle of a document. Document editing software of the 1970s and early 1980s forced you to look at the document’s length, divide that in half, and issue a command to go to the middle page number. With modern-day document editing software, you just drag the scrollbar “elevator” to the middle of the bar, and you are there. Similarly, snap-to grids and alignment guides in drawing tools eliminate the need for users to determine, match, and compute coordinates of existing graphic elements when adding new ones.

- **Make the system familiar.** Use concepts, terminology, and graphics that users already know to make the system as familiar to them as possible, requiring them to think about it less. Designers can use this approach to a certain extent even if the system provides functionality that users have not seen before. One way to do it is to follow industry conventions and standards (e.g., Apple Computer, 2009; Microsoft Corporation, 2009). A second way is to make new software applications work like older ones that users have used before. A third approach is to base the design on metaphors, such as the desktop metaphor (Johnson *et al.*, 1989). Finally, designers can study users to learn what is and is not familiar to them.

- **Let the computer do the math.** Don’t make people calculate things the computer can calculate itself (see Fig. 10.3).
ANSWERS TO PUZZLES ON PAGES 124 AND 125

- The cat is John’s.

- The bear was white, because to have four south-facing walls, the house must be on the North Pole.

- To end up with four liters of water, fill the three-liter bottle and pour it into the five-liter bottle, then fill the three-liter bottle again and pour as much as will fit from it into the five-liter bottle. That leaves one liter in the three-liter bottle. Empty the five-liter bottle, and pour the one liter from the three-liter bottle into the five-liter bottle. Then fill the three-liter bottle again and pour it into the five-liter bottle.

- Let $A =$ the number of cows, and $B =$ the number of chickens. “A farmer has cows and chickens—30 animals total” translates to “$A + B = 30.” “The animals have a total of 74 legs” translates to “$4A + 2B = 74.” Solving for $A$ and $B$ gives: $A = 7$ and $B = 23,$ so the farmer has 7 cows and 23 chickens.

- The Zen student saw three hands up and red dots on both other students. From this information, she didn’t know whether her dot was red or blue. She started out assuming it was blue, and waited. She reasoned that the other students would see her (assumed) blue dot and one other red dot, realize that two red dots were required for all three hands to be up, and quickly figure out that their own dot had to be red. But after a minute neither of the other students had said anything, which told the Zen student that the other students couldn’t figure out what color their dot was, which meant that her own dot was not blue; it had to be red.