2

The History of AI

Intelligence is the main trait that makes humans different from animals. It is what lets humans think about the future and plan activities. It gives humans the ability to communicate complex thoughts. It makes it possible to build and use tools and machines. Because of intelligence, humans recognize patterns and all the other things that distinguish their lives. On the whole planet, only humans are intelligent. Yet, in all of human history, there have been tales of people trying to create other forms of intelligence. In the text that follows, we will look at some of the ways in which people have pursued this quest, from ancient times through to the computer age.

It's worth keeping in mind that each generation thinks of the brain in terms of the highest technology present in its civilization. In earlier times, people did not think in terms of computers because computers didn't exist. People had mechanical technology, and so they thought in mechanical terms. They tried their hand at building mechanical creatures that they hoped would be able to work or perform on their behalf.

The highest technology of the current era is the computer. It follows, therefore, that many scientists think of the brain as a sort of living computer. Today, most of the work on artificial intelligence is focused on computer hardware and software. Researchers are trying to make computers that imitate the way the brain works. Hence, much of what follows focuses on advances in computer science and how it has advanced the field of artificial intelligence research. But first, back to the beginning!

EARLY LEGENDS AND FABLES

The Greeks, Egyptians, and Chinese stand out as having the great civilizations of the ancient world. All of these ancient civilizations had myths and stories about what is now called artificial intelligence. However, most of their stories were told in terms of artificial men. Such artificial men are what we would call robots today.

Take ancient Greece, for example. In Greece, Hephaestus, the god of craftsmen and technology, was said to have made himself metal **automatons**. These automatons, including walking tripods, would help him with his work and bring things back and forth from his home to **Mount Olympus**. According to Homer's epic poem *The Iliad,* Hephaestus also created human-like machines made of bronze to help him move from place to place. One of Hephaestus's creations, Talos, was a giant man made of bronze. Talos was responsible for protecting the island of Crete from pirates and other invaders. He would circle the island three times each day to do so. A similar story tells of a sculptor called Pygmalion who falls in love with one of his creations. This creation is brought to life by the goddess Venus.

Then there's the real-life invention of the ancient Greeks that is so complex that, for many decades, scholars had no idea what it was. When they finally figured it out, they were stunned. It seemed that the ancient Greeks invented a mechanical computer about 2,000 years before anyone else even had an idea that such a thing could be done.

Ancient Egypt, like Greece, had its own mythology that included statues that could feel emotion and think. Yet arguably the most interesting tale of artificial intelligence is from ancient China. More than 2,000 years ago, a tale started of a mechanical man made for the emperor by the craftsman Yan Shi. This mechanical man was said to be able to walk and sing. It also winked at the ladies of the royal court, making the emperor jealous. It acted just like a person. In fact, Yan Shi's mechanical man was so lifelike, the emperor wanted to execute him. Yan Shi then proved to the emperor that his mechanical man was an automaton and not a real human being.

The Antikythera Device: A 2,000-year-old Computer?

A little more than a century ago, in the year 1900, some Greek divers were riding out a storm off the island of Antikythera, just north of the Greek island of Crete. They decided to dive for sponges while they were waiting for the storm to pass. Instead, they found an ancient shipwreck that was later dated to about 100 B.C. During the few years that followed, **archeologists** continued diving to explore the wreck. Then, in 1902, Greek archeologist Valerios Stais (1857–1923) found what looked like a gear stuck inside of a rock. Thinking it was part of an ancient clock, he brought it to the surface. Finding an ancient clock would have been amazing because clocks weren't invented until at least 1,000 years after the ship's sinking.

Decades later, in 1959, British historian Derek de Solla Price (1922–1983) realized something new. The device was really a type of computer called an **analog computer**. It had been built to track the planet positions and moon phases. While a clock would have been an amazing discovery, finding a computer from so many years ago was even more awesome. Also amazing was that scientists found hints about similar machines. These machines, now called Antikythera devices (or mechanisms), were from the same time period as the shipwreck. Far from being a unique invention, it is now thought to have been one of many similar devices made by the ancient Greeks.

What is amazing is that it took another 1,500 years for European workers to make something equally as complex. According to an article in one of the world's best scientific journals, *Nature*, the theories about this device might have been developed before the Greeks. The universe that it

A few thousand years later, in the ninth century, the Arab alchemist Jabir ibn Hayyan (721–815) considered how to create artificial life. Four centuries after Jabir, another Arab scientist named al-Jazari

Figure 2.1 A reconstruction of an Antikythera device is shown on display at the Second International Conference on the Ancient Technology in Athens, Greece, in 2005. It consists of at least 29 gears of various sizes that were made to move simultaneously via a handle.

encoded seems to be line with ancient Babylonian thinking. Thus, the computer is not a recent invention. It is one that can be traced back to ancient times.

(1136–1206) created a device he called a peacock fountain. It used mechanical servants to help guests wash their hands. Al-Jazari also made a band of robots that played music.

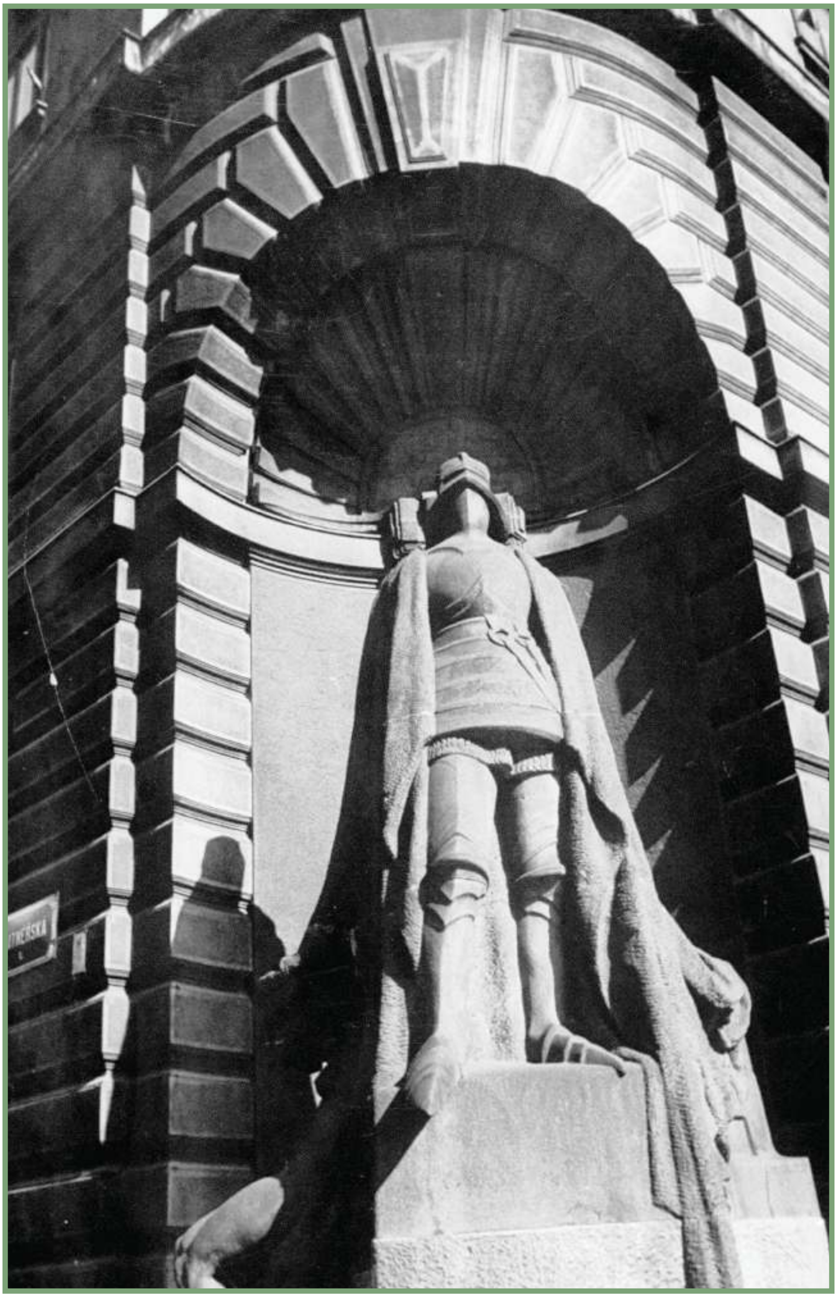

Figure 2.2 The Golem of Prague, the legendary robot defender of Medieval Jews, stands at the entrance to the former Jewish area in the Czech Republic's largest city.

Jewish legends tell about a late sixteenth–century rabbi, Judah Loew ben Bezalel (1520–1609) of Prague, who made a *golem*—a man made of clay that was brought to life through rituals and magic. According to legend, the Golem of Prague was created to protect the Jews of the city from attack. However, it became more and more violent and could not be controlled. It finally was stopped when the rabbi was able to erase the first letter of a word written on the golem's forehead. The rabbi changed the word from *emet* (Hebrew for "truth") to *met* (Hebrew for "death").

What all these creations had in common was that most of them were automata. Automata were mechanical beings that acted but did not think. Although not an automaton, the golem still was not what would be considered intelligent. Even when the golem started acting on its own, it still was not acting intelligently. It behaved like a chaotic animal. Thus, in these fables, there is a part of, but not all of an artificial intelligence. Described in these tales are artificial beings that can move and act on their own, even if they can't think very well. The addition of thinking machines came later, during the industrial age. This was a time when scientists and philosophers guessed about the mysteries of human thought and how it might be copied.

INDUSTRIAL-AGE ATTEMPTS

After the fall of the Roman Empire, Europe fell into a thousand-year slump. The Middle Ages began, a time when progress almost stopped completely. Knowledge in the form of ancient texts was preserved in Europe by Christian monks and Arab scholars. Europeans added little to the sum of human knowledge for nearly a millennium. This began to change in the fourteenth century with the advent of the **Renaissance**. This was a period of 300 years of intellectual rebirth that started in Tuscany, Italy. During this time, people started to look for reasons behind the existence of what they saw in the world. Following the Renaissance was another period called the **Enlightenment**. This was a period even more focused on using scientific principles to try to understand the world.

In an era when people were trying to find reason in the world around them, many wanted to know how the human mind worked. Since people were trying to explain the planets, Earth, the stars, and the animals, it was natural to try to understand the nature of thought itself. Part of this effort was driven by more subtle motives. If people were using their powers of thought to learn about the world, then maybe they should also try to understand their powers of thought. In other words, the mind is the tool a person uses to understand the world. Renaissance scholars and philosophers simply wanted to feel comfortable that they could trust what the mind was telling them.

By the time the Industrial Age emerged, humans were developing and using machines more and more. During this period, the world saw the development of steam engines, steam ships, and locomotives. Humanity was inventing powered devices that could do much of what humans had done in the past with their own strength. It was only natural to assume that human thought was something like a machine and, thus, machines could be designed that would help do tasks that previously could only be done by people.

With this idea as a backdrop, scientists tried to understand the human mind and recreate it in machinery. In the seventeenth century, French philosopher Rene Descartes (1596–1650) took the first step. He described animals as complex machines that could be understood. He was less certain, however, about what made up human thoughts. Around the same time, in 1623, the German scientist Wilhelm Schickard (1592–1635) built the first mechanical calculating machine. He didn't tell many people about his invention and, after his machine was destroyed in a fire a year after it was made, he gave up on it. We wouldn't know about it at all, except for a few letters he wrote that went unfound for more than three centuries.

During the same time, the English philosopher Thomas Hobbes (1588–1679) was thinking along the same lines as Schickard. He concluded that thinking was like calculating. In his respected work *Leviathan*, he said, "reason is nothing but reckoning." In the seventeenth century, the problem was also approached by the German mathematician and philosopher Gottfried Leibniz (1646–1716). Leibniz invented calculus and a number of different mechanical calculators. In addition, he invented the **binary** number system. This system lies at the heart of nearly every computer and computer language in use today. Most interestingly, Leibniz also tried to develop a "calculus of reason." This type of math was based on his belief that it would be possible to use formulas to represent human thought. Leibniz thought it would be possible to calculate equations applied

Figure 2.3 In the early 1840s, Ada Lovelace translated an article by Italian mathematician Luigi Menabrea on the analytical engine, and added her own notes. The notes contain what is considered to be the first written computer program—an algorithm encoded for processing by a machine.

to basic human problems and thoughts. If Leibniz were correct, then we could easily make machines that could think.

Trying to mechanize thought and build machines to solve math formulas were common goals in the following century. This line of work occupied both inventors and philosophers. It reached a peak in the first half of the nineteenth century, primarily thanks to the work of Charles Babbage (1791–1871) and Countess Ada Lovelace (1815–1852). Babbage, a British mathematician and inventor, created the world's most advanced calculating devices while Lovelace developed the world's first programming language to run them. These were the difference engine and the analytical engine. Once built, the difference engine (later called Difference Engine No. 1) could perform simple calculations. It was also great at cranking out tables of numbers. At that time, "computers" were people who performed computations. Babbage's dream was to save computers from the thankless job of calculating many numbers in long math tables. His device would also reduce the number of human errors in these tables.

Though its design was completed, the creation of Difference Engine No. 2 was not completed in Babbage's lifetime. The technology needed to finish it was not around during that era. Following Babbage's plans, a working model of Difference Engine No. 2 was finally built between 1989 and 1991. Its successful creation proved that the basic design was sound.

Running into snags with the difference engine, Babbage regrouped and then (with Lovelace's help) tried to go far beyond his original plans. They designed a machine that would not only perform calculations, but could also be programmed using punched cards to give instructions to the machine. Using these cards, a programmer could build complex sets of instructions. If fully realized, the analytical engine would have been the same as a basic computer. Key to all of this was Lovelace's work. She was possibly the only person to understand what Babbage was trying to accomplish. She was also the inventor of the world's first programming language. As with the difference engine, the analytical engine was never to be built.

Even so, Babbage and Lovelace were on track to making a full-blown computer. In theory, this would have been similar to those being used today. It's tempting to wonder how the world might be different today, had they been successful then: The field of artificial intelligence might well have been advanced by at least a century. In

Figure 2.4 Charles Babbage first conceived the idea of an advanced calculating machine to calculate and print mathematical tables in 1812, as a way to eliminate inaccuracies associated with compiling mathematical tables by hand. Difference Engine No. 1 (*above*) was begun in 1824 and assembled in 1832 by Joseph Clement, a skilled toolmaker and draughtsman. It was a decimal digital machine—the value of a number represented by the positions of toothed wheels marked with decimal numbers.

addition, the first computer would have been powered by steam, and the first programs would have churned their way through wooden and metal gears, instead of electronic components. Instead, the first real computer had to wait for the arrival of the early electronic age.

Before leaving the Victorian Age, it should also be noted that other thinkers had their opinions about the field. British writer Samuel Butler (1835–1902) was the first person to consider that, like animals, machines might evolve. If so, he thought they might someday become smart enough to replace humans.

EARLY COMPUTER-AGE RESEARCH

Babbage died in 1871. After his death, work on computing machines stopped for more than 50 years. Then, in 1941, German engineer Konrad Zuse (1910–1995) invented the first operational and programmable computer. However, Zuse ran into some of the same snags that had caused problems for Babbage. In 1936, Zuse was trying to make a computer with mechanical pieces. Like Babbage, he found that the technology of his day was not up to the challenge. But Zuse kept trying, and within five years was finally able to get the Z3 working.

Instead of **vacuum tubes**, the machine used **electromechanical** telephone parts. Vacuum tubes were invented in 1906, but were not widely used at first. In fact, the German military didn't see the sense in trying to upgrade the Z3 to electronics. Instead, Zuse was put to work making computers that could help design new weapons. Zuse was mostly interested in designing computers for practical purposes anyway. Since Germany was in the middle of World War II, he wanted to help with that effort more than with advancing computer design.

During wartime, on the other side of the English Channel, a British mathematician was busy at work. Alan Turing (1912–1954) was building his own electromechanical computer. It was designed to help break the secret code the Germans were using. The computer he helped design and program was important in helping the Allies win the war. Yet it could be argued that his work in the field of artificial intelligence was even more important than his wartime work.

Turing was the first to come up with the idea of what is now called the **Turing test**. This is a way to decide if a computer is actually

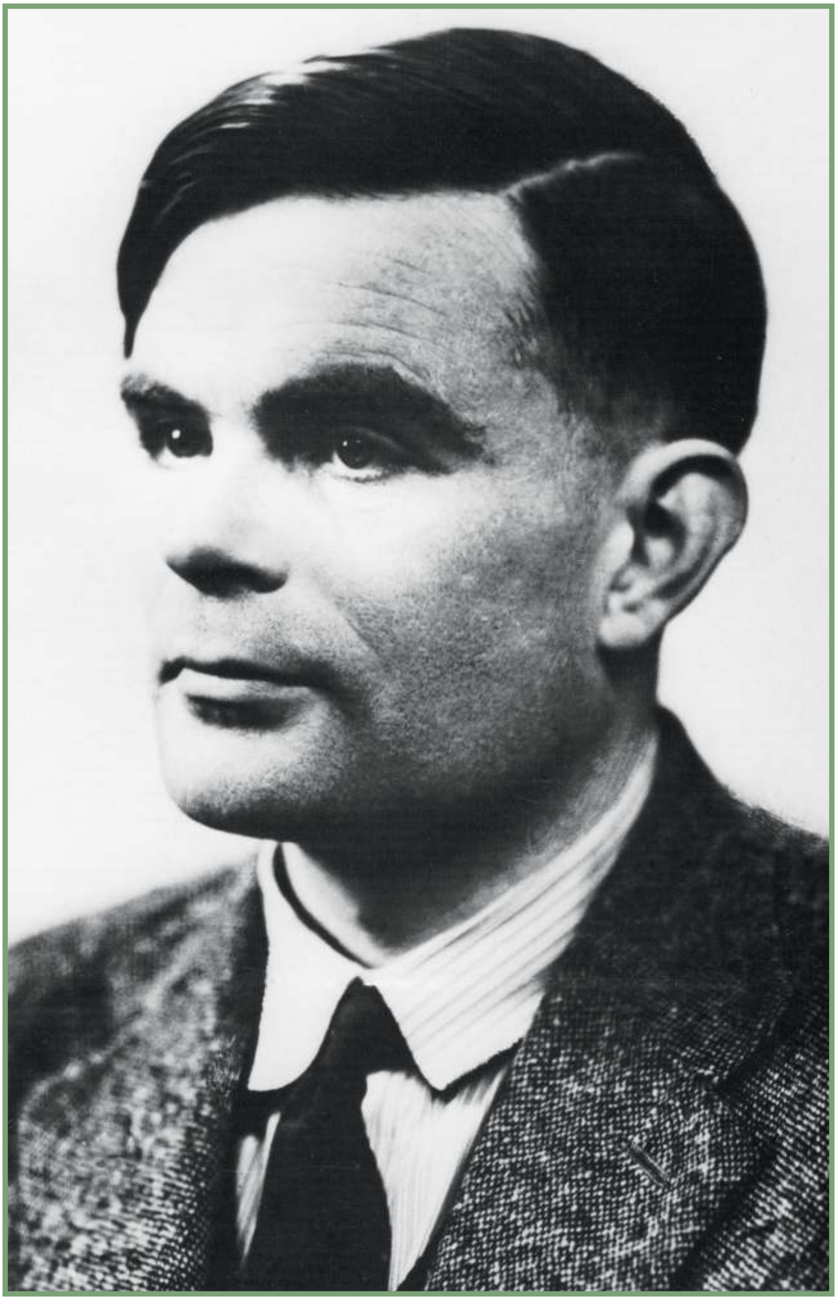

Figure 2.5 British mathematician Alan Turing came up with the Turing test so humans could test a machine's ability to exhibit intelligent behavior.

The Turing Test

Alan Turing found a way to check for a computer's intelligence. A person could communicate through a terminal and guess whether there was a computer or human on the other end. If the person could not tell the difference between a computer and human, then the computer was said to be intelligent. Today, **digital computers** are most often used for these tests.

Turing felt that digital computers could do whatever was asked of them. They could even pass the Turing test, if given enough memory, computing speed, and time. But even though Turing and researchers since his time have felt that computers will one day become truly smart, others have disagreed. These scientists have felt that no matter how powerful and how well it is programmed, a machine can never be truly intelligent.

Some critics have said that intelligence is part of the human soul, and since we can't give souls to machines, we can't make an intelligent computer. Turing's reply was that designing intelligent machines was no different than creating children because, in his opinion, humans in both cases

intelligent. He also came up with the idea of the **Turing machine**. Simply put, this is a machine that will follow a set of instructions according to a particular set of rules. The set of instructions is very similar to what is now called a computer program. This was a new concept in the 1940s, but today just about everybody has a "Turing machine" at home: a computer. The user enters the computer's instructions in the form of a program. Then, the computer follows the instructions according to the rules of its operating system. Turing was the first person to explain the theory behind this whole process.

After Turing, progress in the field of artificial intelligence picked up speed. These advances can be generally divided into one of three categories:

1. ***Hardware***: Engineers worked to develop newer and more powerful computers.

would be acting as instruments of God's will. Another objection is that computers will never be intelligent because they can never have emotions. Turing replied that we have no way of knowing whether any other human really feels emotions the same way that we do. We just assume they do. Thinking the same way, Turing said we shouldn't waste time by worrying or wondering whether computers are either feeling or faking emotions. If a computer acts intelligent, it doesn't matter.

One of the oldest objections about computers being able to be intelligent dates back to the time of Countess Ada Lovelace. The argument is that computers can't be original, since they can't come up with new ideas. Turing felt that enough computing speed and memory helps a computer analyze complex problems. Their answers would even surprise humans. This would show that computers have the ability to be original.

Turing responded to every objection that was brought up, giving his answers for why we should be able to develop intelligent computers. In the time since Turing's death, the debate hasn't changed very much. Turing's answers seem just as relevant today as they were in 1950.

2. ***Software***: Programmers worked to develop new and more powerful programming languages, operating systems, and programs.
3. ***Wetware***: Mathematicians, cognitive scientists, and philosophers worked to develop better theories of how human minds and thoughts work.

For many years, there has been a huge debate over whether or not computers could ever really think. As one example, the pioneering computer scientist and mathematician John von Neumann (1903–1957) gave a lecture in 1949 about computer intelligence. At that time, someone in the audience said that it would be impossible for a computer to think. Von Neumann replied, "If you will tell me precisely what it is that a machine cannot do, then I can always make a machine which will do just that." What von Neumann meant was that he could program a computer to do any specific activity. For example, a programmer can make a computer say, "Hello, how are you?" each time it meets somebody for the first time. A computer can also be programmed to follow any other rule. If we come up with a list of actions we think are marks of intelligence, we can program a computer to follow them. This is a software approach to creating intelligent computers. It assumes that smart programming is all it takes to create artificial intelligence.

Hardware researchers take a different approach. They trust technology to make a machine that is genuinely intelligent. They think that artificial intelligence depends on a computer's speed and processor connections. They also think it depends on memory, the right type of software, or a combination of both. In their view, intelligence will show up on its own when a computer is as complex as a human brain.

Both the hardware and software groups make use of the advances of the wetware scientists. These are the scientists who come up with **hypotheses** about how the brain works and puts thoughts together. One wetware scientist might study how neurons are linked together. Then, a computer engineer would try to build a computer wired together in the same manner. Still another wetware scientist might develop a theory about how neurons communicate with each other, helping humans make decisions. A software engineer would then try to write a program to mimic that process. The work of the wetware scientists supports both of these groups. Without the basic understanding of how the brain and the mind work, it is hard to develop the hardware and software to imitate it.

UNDERSTANDING HUMAN THOUGHT

Since the 1970s scientists have worked hard in all of these areas. They have tried to develop powerful computers and advanced running software. Their goal: to create a genuinely intelligent machine. The problem is that original human thought is still too complex to completely understand. By the 1970s, some researchers were talking about bringing together aspects of human intelligence. Their interest was not in making a computer that could talk with humans. Rather, they wanted to design a program that could imitate human thought in very specific areas. These areas are called *expert systems*.

An expert system is a computer program designed to put into code the same level of knowledge that a human expert would have in his or her subject area. It also follows the same rules a human uses to make decisions. For example, there are expert systems that are used to evaluate EKGs, the electrical patterns coming from the human heart. These signals determine if a person's heart is healthy. In most cases, studies have shown that computer systems are just as good at diagnosing heart problems as human doctors are. The reason for this is that the programmers can interview hundreds or thousands of medical experts. They ask each doctor how he or she goes about making decisions and doing his or her job. With this information, programmers can start to develop rules for how experts evaluate data and situations. Those rules can be programmed into computers.

In many cases, a well-programmed computer can make decisions that are as good as those a human can make. Unlike humans, computers can be trusted to evaluate the same data the same way, every time, regardless of how long they've been working. Simply stated, computers don't get tired. Therefore, in some targeted areas, these expert systems seem to be about as smart as humans.

It's worth going through an example of how this process works. Say you're babysitting and you hear a strange noise. What goes through your mind? There are several possibilities. Maybe somebody is trying to break into the house! It could be that the child is stuck in a closet or one of the rooms of the house. Or he may have fallen down and hurt himself. It's also possible that a branch or a tree fell down, that a pet knocked something off a shelf, or that there was a traffic accident in the street outside the house. There are all sorts of things that can cause a strange sound, so you need to try to figure out what sort of sound you heard. Was it the sound of something breaking, something crashing, someone knocking, or someone pounding? What do you do? Do you call the parents, the police, an ambulance, or nobody at all?

Once you identify the type of sound that you hear, you'll be able to take the next step. So say the sound you heard was a crash. The next question is whether the crash came from inside or outside the house. If it was inside, the child might be hurt or might have broken something, or a pet might have caused a problem. If it was outside, it could be a car crash, a tree falling down, somebody trying to break into the house, or something falling off the house. The next question to answer is where the sound was—inside or outside?

Say that now you think that the sound was from inside the house. You've narrowed down the possibilities quite a bit already, but you still don't know if there's a problem. If, for example, the child fell down, he could be hurt and you might have to call the parents or even an ambulance. On the other hand, if a pet knocked something over on a shelf, then you might just have to explain to the parents what happened. So you might listen for what else you hear. If you hear the child crying, then you'd think that he is the cause of the crash, and that he might be hurt. If you hear the pet running away, then you might think that the pet caused the crash. Once you find out exactly what caused the crash from inside the house, you'll know what you should do.

Think about how you could program a computer to deal with the same situation. You'll probably give it a number of questions to try to answer and, depending on the answer to each question, you'll tell the computer what to do or what to ask next. This is called a **decision tree** and it is at the heart of expert systems. A decision tree for the "strange noise while I'm babysitting" scenario might go something like this:

I hear a strange noise.

1. Is the noise a crash, a break, a knock, or a pounding noise?
2. The noise is a crash. Is it from inside or outside the house?
3. The noise is from inside. Do I hear any other noises?
4. I hear a pet running away and I do not hear the child crying.
5. Therefore, the noise was probably the pet knocking over some furniture, and the child is probably okay.

All of this can be programmed into a computer so that a computer can run through the same questions. For each question, the computer can also be programmed to find the answer. With that answer, it knows what to do next. For example, you could put microphones inside and outside the house to help the computer determine the source of a sound. You can also program a computer to "listen" to the microphones and to tell the difference between a knock, a crash, a breaking sound, and a pounding noise. With proper programming, a computer can find the answer to each question as it surfaces. Then, with each answer, it can be programmed to ask and find an answer to the next question.

The computer can also be programmed to call for help when it can't find an answer or when the information it finds is cause for alarm. For example, if the computer identifies the sound as a crash and it hears the child crying, it can be programmed to call for help. It assumes that the child is hurt. It can also be programmed to call the police if it concludes that it has detected a burglar, or to call the parents if the sound is identified as a tree falling on the house. This type of programming, which could include many other types of scenes, would be a sort of babysitting expert system.

Expert systems can be designed to handle almost any situation like this. They can be used where there is a limited number of issues that can come up and where the questions can be resolved by going through a decision tree. Still, we wouldn't consider these computers to be intelligent. There is much more to intelligence than just following a simple set of rules to make decisions. Even though they seem intelligent in some areas, computers are very limited. A computer is like an athlete who might excel in football but can't play basketball or baseball very well. This means that a computer program might be able to look at an EKG and determine if a person has had heart damage, but it might not be able to see a broken bone on an X-ray. In this way, computers are more limited than humans.

Think of typical adults, for example. Chances are that in addition to their jobs, they can write letters, cook meals, and clean the house without breaking things. They can walk through crowded rooms without falling or running into things, and they can carry on conversations. They can play sports (even if they don't play them very well), and so forth. A typical adult might not be as good as a professional at any single thing, but they can do a far greater variety of things than any computer can do.

Something else that computers can't do very well is deal with questions that don't have simple answers. For example, a computer can easily deal with yes or no questions such as, "Is the noise inside the house?" However, they aren't as good at answering unclear questions such as, “Is the child calling for help because he’s hurt or because he’s joking?” Computers are also bad at making decisions that aren’t expressed by numbers. For example, a computer probably can’t give its opinion as to which painting in a gallery is most beautiful. It can’t say whether a hamburger tastes better or worse than broccoli. It won’t tell which song it likes the best. All of these are questions humans can answer.

Then there are questions that are difficult even for humans to answer, regarding topics of right and wrong. As an example, we know that stealing is against the law. We can program a computer to decide that it’s never okay to steal. We can also program computers to decide that it’s never okay to let someone suffer. But what if someone is suffering from hunger and doesn’t have any money? Is it okay for that person to steal food so that he or she doesn’t starve to death? Most people would be able to come up with an answer for this question, but a computer might not be able to answer it at all. It might see two things that are wrong, and not be able to see that one might be more wrong than the other one.

For a computer to become intelligent, researchers must find a way to teach it to answer these more difficult questions. These questions don’t necessarily have easy, yes or no answers, and they can’t be reduced to numbers. So far, such questions are the ones humans are uniquely able to answer. For computers to answer them, programmers will have to find a way to tell them how to think the way humans do.