1.1 The Lure, and Eeriness, of Machine Life

[00:00] The title of this talk might seem a little odd - to even use the phrase “machine life” might seem a little presumptuous. Can machines actually be alive? This is a conversation that people have had for a long time, and it's a theme that's going to play into a lot of our discussions about how to model the human mind or living minds through the use of machines, primarily computers.

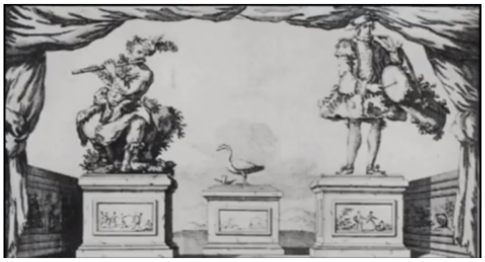
[00:34] When I say this is an old conversation, the ancients actually wrote and talked about the allure of machines that could behave in lifelike ways. In *The Iliad*, by Homer, there is a description, for example, of the god of technology, Hephaestus, who has at his beck and call a group of machines. They're described as golden maidens who are able to help him in building and constructing the shield of Achilles. The golden maidens are described in terms that today we would think of as kind of similar to robots. They’re machines that move and that imitate human beings. It's not clear at all what kind of machines Homer might have been familiar with - *The Iliad* was composed some time around maybe 800 BC. It's not clear what kind of machines he might have been familiar with but he was able to imagine that machines could be constructed that would imitate life.

[01:44] One of the earliest actual examples that we have knowledge of is the machine shown on this slide here by Hero of Alexandria who lived in the first century CE and who constructed a number of very clever machines that could in some ways imitate human behavior. So here he's got a statue which has a fire in it. You light this fire, and it boils a reservoir of water that's inside the machine causing steam to expand and then causing that figure at the right to lower its pitcher of water and douse the statue. So this is a statue that puts itself out. It's a very clever device. The statue itself is no longer in existence, but we have evidence that Hero created or designed such a statue. This is already a kind of intriguing thing. It suggests that people are thinking about machines that can provide the illusion of having purpose or human behavior.

[03:02] Over the years, the technology involved in creating these machines got more and more sophisticated. By the 1400s you were seeing clocks in Europe. This was really the area of technology where sort of fine-tuned machinery made its greatest advances. So, over time, as people started to make clocks, they started to embellish them with life-like figures that could do things like strike the hour. The figure at the upper left here is a clock striking person: they were called Jacks of the Clock in England. And this was made in about the 1400s, and from what I know it's still operational.

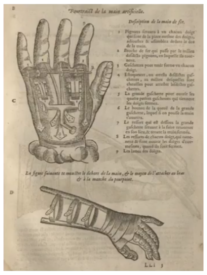
[03:51] Over the next few centuries, art and technology became even more sophisticated. And unlike the kind of public civic architecture of clocks, advanced machines, automata, began to be associated with objects of luxury art made for aristocrats, sometimes, often in fact, embedded with clocks. But at the same time, their appeal was just that they were spectacularly well designed and extraordinarily lifelike.

[04:29] Maybe the high point of the history of automata, of lifelike machines, is given by the picture at the bottom here. The great genius of automaton building was a man by the name of Jacques de Vaucanson who lived in France in the 18th century. And by this time automata were starting to be not just for the aristocracy, but they were starting to be objects of display, almost like special effects. You almost get the feeling that these displays of automata were kind of like summer blockbuster movies or something like that. The automata that are shown in this picture, again none of them are still in existence, but the automata that are shown in this picture are three of de Vaucanson's masterpieces. The one at the left is a flute player and that's an actual flute in its hands. So the machine was actually able to blow air across the top of a flute and get notes out of it. It's kind of an extraordinary work of engineering.



[05:39] Vaucanson's most famous piece was that little duck in the center; people used to come and ooh and aah at this duck. You would place metal pellets in front of it and the duck would lean over and eat the pellets, and you would see its feathers ruffle and then it would excrete. And this was just seen as the most amazing piece of lifelike art ever.

[06:02] The reason I go through these things is because these are instances of machines in people's experience becoming more and more lifelike. I should say humanlike, except in the case of the duck, maybe it's more ducklike. But regardless, these are machines that are becoming more and more imitations of life. That causes people at the same time to start wondering if human beings can be treated as machines. Just as we can say that a machine is sort of like a person, maybe we can also say that a person is a little bit like a machine.



Pourtraict de la main artificielle.

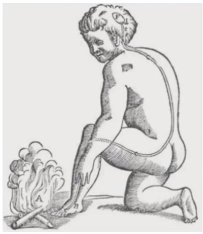
Deſcription de la main de fer.

1 Pignons ſeruans à vn chacun doigt qui ſont de la piece meſme des doigts, adiouſtez & aſſemblez dedans le dos de la main.
2 Broche de fer qui paſſe par le milieu deſdicts pignons, en laquelle ils tournent.
3 Gaſchettes pour tenir ferme vn chacun doigt.
4 Eſtoqueaux, ou arreſts deſdictes gaſchettes, au milieu deſquelles ſont cheuilles pour arreſter leſdictes gaſchettes.
5 La grande gaſchette pour ouurir les quatre petites gaſchettes qui tiennent les doigts fermez.
6 Le bouton de la queuë de la grande gaſchette, lequel ſi on pouſſe la main s'ouurira.
7 Le reſſort qui eſt deſſous la grande gaſchette ſeruant à la faire retourner en ſon lieu, & tenant la main fermée.
8 Les reſſorts de chacun doigt, qui ramenent & font ouurir les doigts d'eux-meſmes, quand ils ſont fermez.
9 Les lames des doigts.

La figure ſuiuante te monſtre le dehors de la main, & le moyen de l'attacher au bras & à la manche du pourpoint.

LLl 3

[06:52] And as people start to think about this, it's actually a rather provocative and worrisome thought, but you start seeing, for example, in the 1500s, the work of a surgeon, a French surgeon named Ambroise Paré. He was a battlefield surgeon, he was interested in making prosthetics. And so, I don't actually know if the diagrams in this page of his book represent things that were in fact actually built. I'm not sure about that. But again the fact that he was modeling and conceiving these things, is quite interesting. To make a prosthetic hand, you are already imagining that the workings of the human hand can themselves be modeled by a machine. In other words, now we're not talking about machines that are lifelike, we're talking about humans starting to be viewed as machine-like.



[07:55] René Descartes, the 17th century philosopher, began actually, he pushed this idea a little further, and he was aware that it was a fairly risky, and to some heretical, idea. But in some ways he was intrigued by the idea of modeling human behavior in machine terms. This is a very famous diagram where he displays his idea of how reflexes work. So, you'll notice that in broad outlines the diagram doesn't look too unlike that Hero of Alexandria statue. The idea is that if a person puts their hand near a fire, then there is some kind of perhaps pneumatic pressure that runs up the lines of nerves to their brain, and causes them, like a machine, to pull their hand away from the fire. That's again, in some ways, a very daring and risky notion of thinking of human behavior as being machinelike. Descartes famously stopped short, however, of thinking of the mind as a machine. But over time even that line started to be broached, and the notion that people could be thought of as machines began to be voiced more and more openly.

[09:25] There's a very famous, but to me rather confusing, book from 1747 by Julien Offray de la Mettrie called *Machine Man*, and I'll read this quote out to you. Again, this is a typical quote from the book, and it's typical in that it's vague and reading the entire book I don't think is of great interest unless you're interested in the history of these ideas. But it's a difficult book and it seems like de la Mettrie is concerned throughout with pushing the boundaries of how far you can call people machines.

> “Simply admit that organized matter is endowed with a motive principle, which alone distinguishes it from unorganized matter... and that in animals, everything is dictated by the diversity of this organization, as I have sufficiently proved. That is enough to solve the riddle of the substances and of man. We can see that there is only one substance in the universe and that man is the most perfect one. He is to the ape and the cleverest animals what Huygens’ planetary clock is to one of Julien LeRoy's watches.”

Again, there's the use of the clock as a metaphor for an intricate mechanism.

[10:46] So far, we've talked about machines being seen as being lifelike and humans being seen as machinelike. And after the 1700s, the lines between the two began to blur even more in literature and in philosophy of science. A wonderful example from the early 1800s is a short story by the German writer E.T.A. Hoffmann. The story is called “The Sandman”. And it introduces this kind of eerie machine named Olympia, a machine woman who is so lifelike as to be strange and disturbing. I have a quote from the story at the bottom of the slide there. It's a fantastic story to read, and all the more fantastic because it begins to point at the anxiety that people feel around machines that can behave like people. The anxiety was somewhat dispelled later in the century when the story was turned into an opera by Jacques Offenbach. And what you're seeing on the screen is a picture of a modern production of *Tales of Hoffmann*, where Olympia the doll sings a song. Offenbach played this for laughs, but the Hoffman story is anything but humorous: it's a very serious and scary story. It's an example of the kind of conversation that began to be felt more and more.

[12:33] The anxiety around machines as people can't be denied, and it's going to be something that in part we're going to deal with as this course goes on. It's reflected in the 20th century in too many to count stories involving machines that act like people, that rebel against people, that fool others into thinking that they're people. The first use of the word **robot** was in a play by the Czech playwright Karel Čapek called *R.U.R.*, standing for *Rossum's Universal Robots*, that was written in 1920. But all through the century there have been numerous examples of machines that make people nervous. Robots, androids, computers, later on in the century as computers arrived, computers run amok, and so forth.

[13:33] And we haven't evaded this fear to this day. This is just one example. Robots can be scary; machines can be scary. I just point to one little thread of this sort of cultural progression in which things like dolls and ventriloquist dummies can be scary. So, there are things like, well, the picture up at the left there, top left is from a movie called *Dead Silence*, with a pretty scary dummy. And recently there have been a series of the *Annabelle* movies with the scary doll. My favorite of these is the one that's at the bottom, a British movie from 1945 called *Dead of Night*. You should really find this movie and watch it. I'll make that an informal assignment for the course: find *Dead of Night* and watch it - it's great. It's really scary, and even if you don't like horror movies generally, it's not gory or disgusting or anything like that, it's just scary. But it features, it's a bunch of stories, and the last story in it features a ventriloquist and his dummy, and let's just say that the relationship between the two of them is not entirely amicable.

[14:49] Dolls and machines and lifelike devices can be anxiety-provoking to us. And that's a theme that in fact is actually going to run through a lot of the discussion that we're talking about. People have an emotional investment in whether humans can be modeled by machines, and how accurate machines can be as models for human behavior. We are going to talk much more about this as time goes on. But just to wrap up this one discussion, we've mentioned the idea that machines can imitate humans and in effect that's the bedrock foundation of the field of computer science known as artificial intelligence. Artificial intelligence is the study of trying to get computers or machines in general to imitate what we would think of as intelligent behavior in humans. Similarly, a bedrock foundation of cognitive science, which we will also be discussing here, is the idea that humans can be modeled adequately by machine programs. We can learn about how humans behave by creating programs or devices that illuminate or imitate their behavior. And finally, there are still frontiers that are yet to be explored involving the integration, the weaving together, of humans and machines. We will be talking about this toward the end of the course, but I mentioned it now just because it's an area of some interest to me.

[16:34] The picture that you're seeing is of an Australian performance artist named Stelarc. And in the picture he's shown wearing a robotic arm that is in fact attached to his arm through electrodes. He can control the arm by, I don't think he does this particular act anymore, but when he was wearing the arm he could control it by muscles in his stomach. And he became facile enough with the arm so that he could do things like write his name, for example. That's an example of weaving together human and machine intelligence that is not quite identical to either of the two more straightforward themes that we talked about so far. But these are going to be the directions that we are talking about - provocative directions - as we move on in talking about cognitive science and artificial intelligence.

1.2 The Turing Test

[00:00] Probably the most famous early idea about equating machine intelligence and the human mind is a notion called the Turing test. It derives from a famous paper by the, I guess you could call him the mathematician and computer scientist, Alan Turing. In 1950, he wrote a paper called “Computing Machinery and Intelligence”, and it was published in the journal *Mind*, a philosophical journal. In that paper, he outlined the idea of an operational test for whether you could say that a computer or machine was intelligent or had a kind of human-like intelligence.

[01:00] Alan Turing along with John von Neumann, with whom his name is usually paired, Alan Turing is one of the two great founding figures of computer science. In 1936, '37 he wrote a paper, which introduced the idea of a Turing machine, which was a mathematical abstraction supposed to represent the idea of a computer. At that point, what people generally meant by a computer was a human being, a person whose job it was to actually do a certain kind of routinized arithmetic in order to compute tables and so forth. There were no computers of the sort that we think of as existing at that point. That is to say there were no automatic computers, there were no electronic computers. But when Alan Turing wrote his paper, he was providing a mathematical abstraction of any kind of computing machinery, and his idea in that paper was to try and formalize the idea of what it meant to be computable by mechanical means, you could put it that way.

[02:24] There was a similar kind of motivating force behind his later 1950 paper, which introduced the idea of the Turing test. He didn't call it the Turing test in his 1950 paper. It's been called that since that time but in his paper, he called it “The Imitation Game”. The basic idea is this: that you put a computer and a person behind two separate doors in two separate rooms, and the doors are closed and you as a judge come up to these two doors and you can type questions by, at the time of the writing of his paper in 1950, it would have been something like a teletype. You could imagine typing over a computer line to both rooms. You can get back textual answers from both rooms. One is a person just answering questions the way a person would. They have no particular instructions except to answer the questions however they want. The other room contains a computer whose job it is to fool you, the judge, into thinking that it is a person. Your job is to tell the two apart. If you can't tell the two apart, if you can't say so here's room number one with the person and here's room number two with the computer, what shall I do? You would not see this as far as you're concerned as the judge, you just see two doors, but behind one of these doors is a person and behind the other is a computer, which I'll just represent as a screen and the keyboard. So your job is to tell these two things apart via the answers that they provide over the computer connection or teletype or whatever. If you can't do that in about 15 minutes or so, then as far as Turing is concerned in this paper, you could say of the computer that it is intelligent. It certainly has a kind of operational intelligence good enough to fool a person into thinking that this is a human.

[04:51] Now, when he introduced this idea in his paper, he actually begins with a scenario that is similar, but not identical to this. He imagines an imitation game played between a man and a woman. So behind one door is a man; behind another door is a woman. You can type questions at both of them. You don't know which room contains which person. So you can type questions at both of them and your job, without loss of generality, you could say that the man's job is to pretend to be a woman and the woman's job is just to answer the question however she wishes. Your job as judge is to tell the two rooms apart. Say, which of the rooms contains a woman answering the questions and which of the rooms contains a man pretending to be a woman. Notice that in this scenario, which by the way, as far as I know, I don't know if anyone has actually done this experiment. Sounds like it could be a fun party game or something like that. But regardless, there are two challenges that have to be met here. One is the man whose job is to successfully pretend to be a woman; the other challenge is that of the judge, whose job is to tell the two apart.

[06:17] You could imagine doing versions of this imitation game test with a lot of other pairings. You could have say, a 20-year-old and a 70-year-old, where the 20-year-old's job is to pretend to be the older person and your job as judge is to tell which of the two rooms contains a genuine 70-year-old person and which contains a young person pretending to be an older person. Or you could do this with different geographic locations: somebody from Alaska and somebody from South Carolina. Or you could do it with different political orientations. There are a whole variety of things. In any of those cases, it might make for a really interesting experiment.

[07:06] In the test that Turing imagines, the job is to tell a person from a computer. Early on in the paper, he provides a sample question and answer that the judge might give, where the judge is asking questions and he or she is getting back answers from one of the two rooms, and the judge's job is to determine whether this might be a computer or a person. So I'll just read it out.

> Q: Please write me a sonnet on the subject of the Fourth Bridge.
> Answer: Count me out on this one. I never could write poetry.

Add a couple of numbers, wait for a while and then a response is printed out. Do you play chess? Yes, I have a king at my K1 and no other pieces. You have only a king at K6 and rook at R1, that's your move, what do you play and so forth.

[08:09] Even in this little exchange, there's something rather clever and charming going on, which is, if you examine the addition problem carefully, you will see that whoever is answering this question, whether it's a person or a computer, whoever is answering this question, gives the wrong answer. Now, you might reason that if you were the judge that superficially, you might think, "Well, that has to be the person, not the computer answering the question because after all, a computer would not get the answer to an addition problem wrong, would it?" Well, remember this is a computer programmed to imitate a human being. Therefore, it might be a computer whose program tells it to make the kind of errors that people are likely to make in addition. In fact, the error that's made here is not a wildly off base error, it might be the kind of error that a person would actually make in adding two numbers. So this is a very cute introduction to the paper.

[09:23] The idea again is that Turing is not saying that if a computer can pass this test, it's a human, nothing like that. All that he's arguing is that as far as we're concerned, we could think of this test as a way of just, in a concrete way, settling the argument about whether a machine could actually be intelligent. If a machine could pass this test, then it would be Intelligent.

[09:54] In the course of the paper, Turing provides a bunch of potential objections to his idea. You get the feeling that in writing this paper in 1950, Alan Turing went around to a variety of people, and talked to them about this idea and got back certain objections and answers, and he wrote these all out and provided his own responses to these objections. As I give this lecture, I'm hoping that you read the paper or have read it before you're actually watching this lecture, and you've already made some internal decisions for yourself about the provocativeness or interest of these various objections. I'm not going to go through all of them just for lack of time, but I'll mention several. People have different favorites or they want to focus on different objections. I'll mention a few of them. The argument from consciousness, Lady Lovelace's objection, and the argument from ESP: all kind of interesting. In fact, all the objections could be things that we could talk more about but I'll just focus on these three.

[11:13] So what about the argument from consciousness? A lot of people would look at Turing's test and say a bit, well sure, you could say that maybe a computer could pass this test, but you wouldn't say that the computer was conscious or had anything like human consciousness in doing this. Turing's response to this, in a way that may or may not convince you, he says something to the effect that, "Well, consciousness is quite mysterious. We don't really know what it involves." But again, if you were having a conversation with the computer and it looked something like the following, and he gives this example:

> Q: “In the first line of your sonnet which reads ‘Shall I compare thee to a summer's day’, would not a ‘spring day’ do as well or better?
> A: It wouldn't scan.
>
> Q: How about a ‘winter's day’? That would scan all right.
> A: Yes, but nobody wants to be compared to a winter's day.
>
> Q: Would you say Mr. Pickwick reminded you of Christmas?
> A: In a way.
>
> Q: Yet Christmas is a winter's day, and I do not think Mr. Pickwick would mind the comparison.
> A: I don't think you're serious. By a winter's day one means a typical winter's day, rather than a special one like Christmas.

[12:31] Now, look, if you were having that conversation with an entity behind the door, you would be willing to attribute, if not consciousness, then certainly a great deal of knowledge to this entity. It's read Dickens, it knows about Mr. Pickwick, and it knows that Christmas is a special winter's day, and it knows about the rhythm of lines in poetry. It feels like a very natural conversation. In fact, bordering on an interesting conversation, the one he was having with this entity.

[13:12] So Turing's argument is well, if you could have a conversation of this kind, then for whatever it's worth, you'd probably be willing to attribute intelligence and perhaps consciousness to that entity. It might be very difficult to tease apart whether the entity really did have consciousness or not. But certainly, in an offhand and formal way, you would be willing to attribute consciousness to it. If I say this may or may not convince you as an argument, but that's the response that Turing gives.

[13:50] In the case of Lady Lovelace's objection, that's an interesting objection. He quotes a paper that was written in the early 1840s by Countess Ada Lovelace. Countess Ada was a colleague, a collaborator of the British mathematician, Charles Babbage. Charles Babbage, this a fascinating story and I only wish I had more time to go through the entire history of it because it really is an amazing story. Charles Babbage was an early 19th century genius, a mathematician. He had many many interests. Over his life, his overriding desire, his obsession, became to build something which he called the Analytical Engine. Analytical engine being a mechanical device that could automatically do a wide, it in fact has a strong resemblance to a general purpose computer.

[15:07] If you read about Babbage's life and you read some of his notes, it's very clear that he had many of the pioneering ideas about how a computer might work, and what a computer could do although in his vision, he was thinking of this as a mechanical, perhaps steam-driven entity rather than of, naturally this being in the early 1800’s, rather than being an electrical device. Nonetheless, his ideas about computing were quite amazing. He never did actually get to build the analytical engine because the engineering of the time was not up to the cost and precision that would have been required to build a working analytical engine. Although, there's more to say about that as well.

[16:03] In any event, Babbage worked with Countess Ada Lovelace, who was the daughter of the famous poet, Lord Byron. She actually co-wrote, along with a very cogent Italian writer named Luigi Menabrea, she co-wrote a large paper, a thorough paper, on the idea of the analytical engine, how it would be built and how it would be programmed. It is an amazing paper even not considering that it was written in the early 1840s. It's just a thoroughly amazing paper. You can find it on the web, it's well-worth reading to see how somebody who was just pioneering many of these ideas, who was thinking about these things for the first time really in human history, was starting to think about the notion of automatic computing.

[17:07] In the course of that paper, Countess Ada says something very interesting and something that has been echoed since the electronic or digital computers have been made. Namely, she didn't quite use this language, but this is essentially what she was saying: a computer can't do anything on its own, it can't think on its own, rather, it can only do those things that the programmer has told it to do. So it doesn't have any innovative intelligence. It just does the things that a program could do. That's a very interesting idea, and that's one of the arguments that Turing alludes to in discussing his own thought experiment of the Turing Test.

[17:56] There are different ways of responding to this objection. One, speaking as a programmer, I can tell you, and you may have had this experience too if you're a programmer, speaking as a programmer, I can say that it is often the case that a program does things that I suppose in the sense I told it to do but were very unexpected to me. In other words, yeah, the program is doing what the programmer told it to do, but the programmer doesn't always know what he or she is telling the computer to do. Maybe more powerfully, you could certainly see the possibility of a computer program learning from experience over time. In which case, depending on the experience that the program has, it learns to do things that are well beyond anything that the programmer might have initially thought to put into the program. So these are potential responses to Countess Ada's very thoughtful and interesting objection. That's not the only reason to read Countess Ada's paper: it's a brilliant paper and has many other wonderful ideas and predictions in it as well.

[19:20] Finally, I should mention the objection from ESP. I don't have a slide for that but Turing was responding to experiments that had recently been done by a researcher named J. B. Rhine, who was working I believe at Duke, and who had done a variety of these experiments, I'm not positive I'm getting this right, but it's something along the line like telepathy: where one person would look at a card with a symbol on it and another person across the room would try to guess what symbol the first person was looking at. And according to Rhine, there were people who were able to guess the observed symbol at a much higher rate than chance. I'm not going to go into arguments about these experiments here, suffice it to say that those experiments of Rhine have never been replicated, certainly not satisfactorily replicated. Frankly, I find the results very dubious. But if you're reading Turing's paper and you see him allude to this objection from ESP, you might think that he's being facetious but he's not: he's just keeping an open mind, like a good scientist and saying, "Well, if these experiments are on the level, then there may be a great deal more involved in thinking or intelligence than we've been ready to acknowledge this far." So he's just saying, if that's true, then perhaps there's something going on that we could not mimic with a computer. In retrospect, I don't think it's a very powerful objection.

[21:16] Well, what about the Turing test itself? Turing made a prediction at the end of his paper that he felt that the test would be passed by a computer program around the year 2000. As I'm speaking now, it is the year 2017 and no program has come close to passing the Turing test. There is no program out there such that you could do this kind of experiment, where you put the program behind one door and a person behind another door, and the program could reliably fool a judge into thinking that it's a person after about 15 minutes or so. Nothing close to it. Which leaves us with a rather interesting question: if the Turing test cannot be passed, why? What was Turing missing? Maybe one of those objections that he was referring to is more powerful than he gave it credit for being. In other words, maybe there is a deep philosophical reason why no computer program could ever pass the Turing test.

[22:31] That's why we, the community of computer scientists and researchers, have not managed to come up with a successful program in 67 years since Turing wrote this paper. As an aside, it's not that this has been the main goal of the computer science community in those intervening 67 years, for a long time, there was a general lack of interest in actually trying to program a computer to do this kind of thing. You might say that, "Well, people have not really worked very hard on this problem, so maybe it's not that surprising that no program has passed the Turing test." Okay. There's some merit to that, and a lot of computer scientists, and researchers, and artificial intelligence do not feel that the Turing Test is a terribly meaningful test at least as far as their own research is concerned.

[23:36] But let's leave that aside and just say that, in fact, no computer program has passed the test, maybe it can't be passed. On the other hand, if it can be passed, look at the other side of the question: if the Turing test can be passed, what are we missing? Why has it taken so long? So that's a kind of interesting pairing of questions. If the test can be passed, then what's taking us so long? If the test can't be passed, what is the reason for it? We don't have satisfactory answers to either of those questions. But we do have programs that in one fashion or another, do kind of pass little local Turing tests. They're not general Turing tests of the kind that the paper imagines, but they're still interesting.

[24:38] The Watson program, which famously did so well on the Jeopardy quiz show, is one example. And you're probably familiar with a teeny little Turing test that you see on the web all the time: the CAPTCHA program, which is, in essence, a little Turing test. You are now trying to pass the Turing test to show that you are a human being, not a computer. So this is a little test to see if you can read a couple of poorly written words. This would be a task that computer programs would have some difficulty doing, but people are able to do easily.

[25:21] There's a lot of interest even if you think about this kind of task. First of all, as you may have noticed, computers get better and better at passing these tests. So there's an arms race involved. Over time, the CAPTCHA questions have to become more challenging for the person to the point where, at least in my experience, sometimes I'm trying to get onto a web page and I get a question like this, and I don't think I can read the word that they're telling me to read. So it's getting to the point where I'm not a sufficiently good human being to actually pass the CAPTCHA test. But these are the kind of tasks that we'll also return to in discussion as well in this course. There are other kinds of small-scale Turing tests like whether a computer is able to write music, for instance, that might convince you that it was written by a human being. Other kinds of things like that; all quite interesting.

[26:35] Finally, I should not leave this topic without mentioning a very famous response to Turing's paper written, I think around 1980, by the philosopher John Searle. He posed a counter thought experiment to Turing's test and his thought experiment is called the “Chinese Room” experiment. The idea behind this experiment is you imagine, here's his scenario. You have a room. I'll just draw the boundaries of the room here. So, and there's the outside of the room and, on this side on my side of the screen here is, the inside of the room. And there's a guy on the inside of the room and he has a large rule book. This is my highly imperfect drawing of a large rule book, and lots and lots and lots of scrap paper in a big basket. So lots and lots of scrap paper and a pencil; as much paper and as many pencils as he could possibly need. There is a little input slot in the room and a little output slot and here's the idea. Here is this guy. Let us assume that he, like me, is a native English speaker and does not speak a word of Chinese. Okay? This entire room is going to be playing the role of the computer in the Turing test. It's going to be answering questions in Chinese and the input slot, therefore, has questions written in Chinese characters.

[28:40] The guy on the inside of the room does not understand a single symbol of Chinese. However, he's able to take these input characters and then use this rule book, which has a bunch of rules that tell him how to manipulate other kinds of characters in response to these characters. He can write down any notes that the book tells him to write down on the scratch paper. So he has all the scratch paper he needs. He follows very explicitly, to the letter, the rules given in the rule book, and the rules are such that they tell him eventually to get another scrap of paper and write on it a certain number of characters, which are totally meaningless to him, and then he passes those characters out the output slot.

[29:38] In other words, think of the entire behavior of the room. It gets Chinese questions in, this guy who does not understand Chinese is taking the rules in this book and using them to manipulate various marks on paper and eventually write some other Chinese characters. And presumably, the word book is written in English so he can understand the rules. But the rules tell him to write down certain squiggles and squaggles that he doesn't understand and then he passes those, to him meaningless symbols, out the output slot. Searle argues that conceivably this room might pass the Turing test in Chinese if the program were good enough to mimic the behavior of a Chinese speaker. So Searle says, well, what would happen if, suppose this room did pass the Turing test? Where is the knowledge of Chinese in here? What knows Chinese here? It's certainly not the guy: he doesn't know a word of Chinese. The book doesn't know Chinese. You can't say that a book knows anything. The book doesn't know Chinese. Certainly the scrap paper doesn't know Chinese. So what is it? Where is the knowledge of Chinese in this entire system? In this brief summary, probably as far as Searle is concerned doesn't begin to do justice to his argument, but basically the argument is that you could have a mechanical entity that passes the Turing test, but it would not have what we would call consciousness. It's just a set of mechanical rules for manipulating symbols.

[31:32] Like Turing, Searle includes a variety of objections to his Chinese room scenario that he has heard from various people in presenting this thought-experiment. People responding to Searle's thought experiment and saying, no, I think you're presenting an unfair thought experiment to the purposes of artificial intelligence. So I would leave to you the task of reading Searle's argument, the very pleasurable task of reading Searle’s paper, and going through the arguments that are made in response to the Chinese room and thinking about his responses. There are many wonderful responses in the literature as well. People who have written since Searle's original paper. There have been many people who have responded to the Chinese room argument, and I have some favorites among those counter responses. But I won't go into them here and would just leave it to you to read Searle's argument.

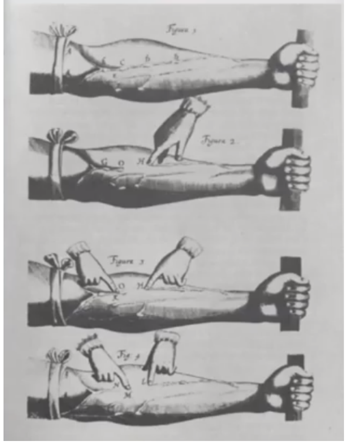
[32:49] Let's leave it with this position. We have a, as yet unmet, challenge from Alan Turing for machine intelligence. We have a provocative, but certainly not universally agreed to, counter argument from John Searle saying that, even if a computer could pass the Turing test, and none have done so so far, but even if a computer could pass the Turing test that would not be, for us, a very important or meaningful step in as much as this would not indicate that the program passing the Turing test has actual consciousness. What we would call meaningful intelligence perhaps. So we'll leave it at that, and as we continue to discuss machines and minds, we'll talk about what things have been accomplished in computer programming to imitate human intelligence.

1.3 The Computational Metaphor of the Mind

[00:00] In the history of science, metaphor has been an extremely powerful and abiding intellectual tool. People often understand one element of the world by making an analogy, a rich analogy, to some other aspect of the world. And, it's fair to say that over time, as our technology grows, as we have more and more instances of technology around, we're able to use those instances as the sources of metaphor for still other phenomena in the world. What I'm getting at is that in the history of cognitive science, in particular, there's been one reigning metaphor. One sort of, you know, foundational metaphor that has motivated a lot of the work in cognitive science; you might, in some sense, say just about all of the work in cognitive science over the past 60 or 70 years. And that is the idea that the human mind or animal minds can be modeled by a digital computer. So we'll talk about that metaphor.

[01:24] But actually I want to introduce the idea by talking about an earlier metaphor from the history of science. Science is filled with these things, that is, people continually sort of look to understand one phenomenon by linking it systematically to another. The example I have here is from the early days of the Scientific Revolution. The portrait is of William Harvey, who was an English physician, and I believe he was actually the Royal Physician to King James the first. And the page that you see is a page from a book that he wrote in the early 1600s about the workings of the heart.

[02:10] The main idea behind this book was a metaphor, was to propose that the heart acts like a pump. And that was a very innovative metaphor at the time because prior to Harvey's book, the sort of raining metaphor of the heart was that it was kind of a brewery. That was sort of the metaphor that would have been favored by the ancients, by writers like Galen, the classical physician. That is they felt that the heart's purpose was to purify blood and that it had two parts, two chambers. And in one chamber theret was impure blood and then somehow that was transferred to the other chamber of the heart where it became purified. That was sort of the portrait of the heart.



[03:00] Harvey proposed a different metaphor. He said no, the heart is not like a brewery, it's more like a pump. And he performed a variety of new experiments to show that this was a rich metaphor, and you could take it very far for understanding what the nature of the heart was. So this is an early example. It's not by any means the first example of the use of metaphor in science, but the history of science is just filled with things like this.

[03:32] I put up a few pictures from the web just to illustrate other chapters in metaphor from the history of science. At the upper right there, there's the idea that the solar system is kind of like a clock. So in the early investigations of the solar system by Newton and by Galileo and then prominently by Newton and then by successors, there was a kind of metaphor that the solar system works kind of like a clockwork, a giant sort of well-tuned mechanism. And as I say, that's an example of the use of technology, a clock, to motivate a metaphor for understanding the natural world. But then once you've got that idea of the solar system as a kind of working system, it could be used as a part of a metaphor that is the foundation of a metaphor for a still later chapter in the history of science, where people could kind of understand the structure of the atom as being sort of like a tiny little solar system. And students of electricity are quite familiar with the common use in elementary electrical circuits of imagining that voltage, that is to say, potential current and resistance can be treated analogously to water pressure and water flowing through pipes.

[05:16] I should mention that none of these particular metaphors is exactly correct. They're helpful, and a good metaphor is defined by its being helpful. It helps you to ask new questions, to propose new directions for research, to conduct new experiments, the way that William Harvey did. But that doesn't mean that a metaphor has to be a sort of perfect match. And in fact in none of these cases, is it a perfect match.

[05:50] The solar system is not quite like a clockwork. It was found in the 1990s, for example, that the orbit of Pluto, which I still wish I could call a planet but I guess technically it isn't, but the orbit of Pluto actually exhibits chaotic patterns in its movements in and out of the orbit of Neptune. So the solar system is not quite a sort of regular clockwork nor of course is the atom a little solar system. Anyone who's studied quantum mechanics and chemistry knows that there are limits to how you can treat the atom as a tiny little solar system. And similarly in electric circuits, you don't want to push this metaphor of water pressure and narrowness of pipes and so forth. You don't want to push that too far. It's pretty hard, for example, to imagine how you can treat a capacitor in this kind of model. So in any event, none of these metaphors are perfect. Metaphors aren't intended to be perfect.

[06:59] In Cognitive Science, we have a kind of traditional metaphor, which is not in itself uncontroversial. Where historically, what we want to say is that the mind is kind of like, in the strongest versions people might even make it a little more, a full-throated defense of the mind is a computer. I'm personally more comfortable with saying the mind is like a computer in some ways. In any event, the innovation here was that once computers arose, that is, once there were computers to use as a foundation of this metaphor (starting in the late 1940s, early 1950s and thereafter) people began to look at the operation of the computer as an information processing device. And they were willing to sort of go out on a limb with a new metaphor and say that the mind, the human mind, is itself like a computer; it's an information processing device.

[08:17] This was seen at the time as a response to what was then the reigning tradition in American psychology, which is behaviorist psychology. So the behaviorists argued again, there were a variety of, you know, philosophies that could go under the name of behaviorism. At its strongest, the behaviorists might argue that you really should not talk about things like ideas or concepts or beliefs or desires or anything internal to the mind because those things cannot be directly seen or measured. And since they cannot be directly seen or measured, they are scientifically illegitimate. So the behaviorist at their strongest, had this kind of rigid view that you should not be able to talk about the internal workings of the mind. The organism should be treated essentially like input output devices, like black boxes. If you give them a certain kind of input, you get a certain kind of output, so forth.

[09:28] So the cognitive revolution in Psychology was based on a metaphor, people were now able to say, no we can talk about things like ideas and concepts and beliefs and desires if we say that the mind is in fact like a computer. So just as we can study the behavior of a computer, and we can write programs for it, similarly, we can use those programs as a metaphor for the operations of the mind. If you take the metaphor really seriously, and by seriously, I mean sort of at its most intense, you actually treat the metaphor as accurate, not just a little bit of poetic license, then you could make the argument that the software of the computer is rather like the operations of the mind within the brain. That is, software is to hardware as mind is to brain. The brain runs the mind as software much as a digital computer runs its software to perform all kinds of different actions.

[10:54] That's a very strong version of the computational metaphor of mind. There are far weaker versions and there are versions that allow for a lot of hand waving that, in some respect, you can treat the mind as operating like a computer or you can write programs for computers that illuminate certain operations of the mind. Those are, again, much gentler and usually less controversial versions of the computational metaphor.

[11:27] The most intense version is this one, and if you take that seriously, then there are a couple of things that sort of follow from it. And these are again, controversial conclusions, but they follow from the sort of most direct interpretation of this metaphor. First, if software is to hardware as mind is to brain, then if you want to understand the mind, you don't really have to understand the brain to do it. In other words, you don't have to explain the mind at the level of Neuroscience. Why would that be true? Well, if for those computer scientists who may be watching this, you know well that if you're a computer programmer, you don't really have to have a very deep idea about the hardware of the machine that you're writing for. Unless you're writing in a very, you know, sort of low-level language like machine code or assembly code or something like that. But if you're writing in a high-level language, you don't really have to know much, perhap, you don't know anything, about the hardware of the machine that is going to run the program that you're writing.

[12:45] So you can study the algorithms and the computational ideas in software without knowing much of anything about hardware, and computer science students routinely take courses in algorithms, for example, where hardware is virtually never mentioned. If you understand the idea of a quicksort algorithm, you don't have to understand anything in particular about the specific machines on which that quicksort algorithm is going to be run. You can talk about quicksort as an entity in its own right. You can talk about how long it takes to run, whether it's a good idea, whether it's an efficient algorithm, what its limitations are, and you don't have to talk about hardware at all.

[13:35] Similarly, if you take this metaphor of mind seriously, then to talk about the mind in software terms, you need not talk about the brain. You could talk about, for example, the process of language acquisition or visual perception or a variety of things without necessarily referring to the implementation of those ideas in neurons, in human neurons in the brain. That as I say is a rather controversial interpretation, and I think it's fair to say that most cognitive scientists, the great majority, do not hold to that interpretation anymore. However, this was a version of the computational metaphor that played an especially powerful role in the early years of cognitive science.

[14:30] A second thing, which is mentioned on this same slide, is that if you take this metaphor seriously - then just as computer scientists study algorithms and study things like arrays or lists or data structures or whatever, then in the same vein you could treat mental representations as the data structures and you could study those data structures, however they're implemented: parse trees, symbols, rules, collections of rules, sets of pixels as images, all kinds of different representations. So you could study those representations and treat them as themselves the objects of study, just as you would study data structures and algorithms in the realm of computer science. Again, that's a very interesting idea, that now where the behaviorists would rule those things out of court, that is, you're not allowed to study things like parse trees and rules because we can't see them and we can't measure them directly. Now, since we can write programs which can mimic human behavior, we are allowed to talk about the questions of whether humans at least behave as though they are running sets of rules or working with mental images or using context-free grammars as the basis of their language and so forth.

[16:15] Okay, so a corollary of this, the sort of philosophical support for this idea of the computational metaphor goes by the name of functionalism. Actually, I think there's a few different uses of the word functionalism in philosophy, but this is a particular one that applies to cognitive science. So I'm just going to read off the slide here:

> Functionalism is generally summarized as the notion that mental states are characterized according to their causal roles in a system of mental states. In particular, it doesn't matter in what physical substance these states happen to be embodied. There is a resonance here with the notion of a computer program; it doesn't matter whether that program happens to be rewritten for a Macintosh or Cray or whatever -- the essential program remains the same.

So the idea of functionalism is that you can, you can study things like the mind as a working system of rules, states, computational elements. And what's important is how those things work together as a system, but it doesn't particularly matter what physical machinery or substrate those things are implemented in. Again, this is just another sort of fancy philosophical way of describing the strongest version of the computational metaphor of mind.

[17:53] As we talk about the variety of research efforts in cognitive science, we will also talk about the strengths and limitations, and possible futures of this overall computational metaphor. The story of the computational metaphor of mind is not by any means finished, but it's been an interesting saga, and we're going to be talking about both the criticisms of that idea and where that idea has had its own kinds of successes.