i) Gastrulation.
ii) Polyogonally symmetrical structures, e.g., starfish, flowers.
iii) Leaf arrangement, in particular the way the Fibonacci series (0, 1, 1, 2, 3, 5, 8, 13, ...) comes to be involved.
iv) Colour patterns on animals, e.g., stripes, spots and dappling.
v) Patterns on nearly spherical structures such as some Radiolaria, but this is more difficult and doubtful.

I am really doing this now because it is yielding more easily to treatment. I think it is not altogether unconnected with the other problem. The brain structure has to be one which can be achieved by the genetical embryological mechanism, and I hope that this theory that I am now working on may make clearer what restrictions this really implies. What you tell me about growth of neurons under stimulation is very interesting in this connection. It suggests means by which the neurons might be made to grow so as to form a particular circuit, rather than to reach a particular place.

4 The Turing Test

"Intelligent Machinery" contains the earliest description of (a restricted chess-playing form of) what Turing later called the "imitation game" [95, p. 23] and is now known simply as the Turing test. Turing's "Computing Machinery and Intelligence," published in 1950, introduced the famous version of the test [96].[70]

4.1 The Form of the Test

In the 1950 presentation of the test, the interrogator or judge communicates (by keyboard) with both a computer and a human being; apart from this no contact is permitted. The interrogator's task is to find out, by question and answer, which is the computer [96, p. 434]. The computer attempts to evade identification. Turing said that the best strategy for the human foil is probably to give truthful answers. He presented the following sample dialogue:

JUDGE 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?
COMPUTER It wouldn't scan.
JUDGE How about "a winter's day"? That would scan all right.
COMPUTER Yes, but nobody wants to be compared to a winter's day.
JUDGE Would you say Mr. Pickwick reminded you of Christmas?
COMPUTER In a way.

[70] For a discussion of the differences between this and Turing's 1952 presentation of the test, see [27] and Copeland's introduction to Turing's "Can Automatic Calculating Machines Be Said To Think?" [24].

JUDGE Yet Christmas is a winter's day, and I do not think Mr Pickwick would mind the comparison.

COMPUTER 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. [96, p. 446]

4.2 Scoring the Test

In the 1950 paper Turing first described an imitation game involving an interrogator and two human subjects, one male (A) and one female (B). The interrogator communicates with A and B by keyboard. The interrogator is to discover, by asking questions, which of A and B is the man; A's aim is that the interrogator make the wrong identification. Turing then asked "What will happen when a machine takes the part of A in this game?" [96, p. 434]. The point of this new game is to determine whether or not the machine can "imitate the brain" [98, p. 464].[71] To assess the computer's performance, we ask:

> Will the interrogator decide wrongly as often when the [computer-imitates-human] game is played ... as he does when the game is played between a man and a woman? [96, p. 434]

If the computer (in the computer-imitates-human game) does no worse than the man (in the man-imitates-woman game), it passes the test.

The function of the man-imitates-woman game is often misunderstood. For example, according to Hodges, this game is irrelevant as an introduction to the Turing test — indeed, it is a "red herring" [48, p. 415]. However, the man-imitates-woman game is not intended as an introduction to the test; it is part of the protocol for scoring the test.

4.3 Did Turing Propose a Definition?

He is widely said to have done so. For example:

> An especially influential behaviorist definition of intelligence was put forward by Turing [96] in [6, p. 248] (Block).

> The Turing Test [was] originally proposed as a simple operational definition of intelligence .... [42, p. 115] (French)

> [Turing] introduced ... an operational definition of "thinking" or "intelligence" ... by means of a sexual guessing game. [48, p. 415] (Hodges)

[71] Some commentators suggest that Turing's intention is to describe a game in which the computer must imitate a woman. Textual evidence against this suggestion is given in [27].

Those who characterize the Turing test as a definition lay Turing open to easy objections. This is because it is all too conceivable that an intelligent, thinking entity could fail the test, for example if it were distinctively non-human in its responses.

In fact, Turing did not propose a definition of "thinking" or "intelligence." Definitions are standardly expressed as necessary and sufficient conditions. In his 1950 presentation of the test, Turing emphasized that passing the test (by producing behavior that resembles human intellectual behavior) is not a necessary condition for thinking. He said that machines may "carry out something which ought to be described as thinking but which is very different from what a man does" [96, p. 435]. And in his 1952 presentation of the test, in a radio discussion (with, amongst others, Max Newman)[72], he explicitly denied that he was offering a definition:

> I don't want to give a definition of thinking, but if I had to I should probably be unable to say anything more about it than that it was a sort of buzzing that went on inside my head. But I don't really see that we need to agree on a definition at all. The important thing is to try to draw a line between the properties of a brain, or of a man, that we want to discuss, and those that we don't. ... I would like to suggest a particular kind of *test* that one might apply to a machine. You might call it a test to see whether the machine thinks, but it would be better to avoid begging the question, and say that the machines that pass are (let's say) 'Grade A' machines. ... Of course I am not saying at present either that machines really could pass the test, or that they couldn't. My suggestion is just that this is the question we should discuss. It's not the same as "Do machines think," but it seems near enough for our present purpose, and raises much the same difficulties. [100, pp. 466–467]

4.4 Fiendish Expert Objections to the Turing Test

There are many of these, all of the form "An expert could unmask the computer by asking it ... " For example, it is said that questions designed to elicit the various human irrationalities claimed by theorists such as Wason, Nisbett, Tversky, and Kahnemann could be used to distinguish the computer from the human foil.[73] Unless specially programmed to duplicate the particular human irrationality, the computer would be easily detected.

However, Turing anticipated this type of objection by ruling out the use of expert judges. He stated in 1952 that the judges "should not be expert about

[72] The full texts of this and of Turing's 1951 radio broadcast "Can Digital Computers Think?" were published for the first time in Copeland's [24]; see also [27].

[73] For example, by Lenat in his paper at the 2000 Loebner Turing Test Competition at Dartmouth College.

machines" [100, p. 466] and in 1948, with respect to the early chess version of the test, that the judge should be a "rather poor" chess player [95, p. 23]. It seems highly likely that, if Turing were writing now, he would exclude interrogators who are expert about the mind.

4.5 Shieber's Criticism

In a vigorous critique of the Turing test, Shieber ascribes to Turing the view that "any agent that can be mistaken by virtue of its conversational behavior [for] a human must be intelligent" [83, p. 70]. This leads easily to an objection. A computer's performance may be a lucky fluke, a thoroughly atypical performance, in the fashion of a first-season football star whose performance subsequently regresses to the mean. Why should we conclude, from the fact that the judges are persuaded by a single successful test, that the computer thinks?

However, Shieber's interpretation of Turing is unsympathetic. A more charitable view is suggested by Géraud de Cordemoy's anticipation in 1668 of Turing's test. De Cordemoy was a Cartesian who wrote about the problem of distinguishing thinking from non-thinking things. He remarked:

> [Concerning] ... bodies ... who resemble me so perfectly *without* [i.e., externally] ... I think I may ... establish for a Principle, that ... if I finde by all the experiments I am capable to make, that they use speech as I do, ... I have infallible reason to believe that they have a soul as I. [36, pp. 13–14]

By insisting that a candidate thinker perform satisfactorily in "all the experiments" that we are "capable to make," de Cordemoy allows for the fact that the results of some experiments may be misleading — a machine that happens to pass one Turing test, or even a series of them, might be shown by subsequent tests to be a relatively poor player of Turing's imitation game. De Cordemoy's position is obviously not susceptible to Shieber's objection. There is no reason to believe that Turing is any more vulnerable to the objection: Turing's position as he describes it is entirely consistent with the de Cordemoy-like view that the result of any given test is defeasible and may be disregarded in the light of other tests.

4.6 The Shannon-McCarthy Objection

In 1956 Shannon and McCarthy formulated the following objection to the Turing test:

> The problem of giving a precise definition to the concept of "thinking" ... has aroused a great deal of heated discussion. One interesting definition has been proposed by A. M. Turing: a machine is termed

> capable of thinking if it can, under certain prescribed conditions, imitate a human being by answering questions sufficiently well to deceive a human questioner for a reasonable period of time. ... A disadvantage of the Turing definition of thinking is that it is possible, in principle, to design a machine with a complete set of arbitrarily chosen responses to all possible input stimuli ... Such a machine ... merely looks up in a "dictionary" the appropriate response. With a suitable dictionary such a machine would surely satisfy Turing's definition but does not reflect our usual intuitive concept of thinking. [82, v–vi]

This objection has subsequently been rediscovered by a number of authors, and is commonly but mistakenly believed to have originated with Block [5].

How might Turing have responded? The 1952 radio discussion provides an indication:

NEWMAN It is all very well to say that a machine could ... be made to do this or that, but, to take only one practical point, what about the time it would take to do it? It would only take an hour or two to make up a routine to make our Manchester machine analyse all possible variations of the game of chess right out, and find the best move that way — *if* you didn't mind its taking thousands of millions of years to run through the routine. Solving a problem on the machine doesn't mean finding a way to do it between now and eternity, but within a reasonable time.

TURING To my mind this time factor is the one question which will involve all the real technical difficulty. [100, pp. 473–474]

Storage capacity and processing speed are crucial. The Shannon-McCarthy objection establishes only that the proposition "If x plays Turing's imitation game satisfactorily, then x thinks" is false in *some possible world* — a world very different from the actual world, since in the merely possible world the machine runs through the routine in a reasonable time. But there is no reason to believe that Turing was claiming anything more than that this proposition is *actually* true.

4.7 French's Objections: Associative Priming and Rating Games

The associative priming objection runs as follows. In word/non-word recognition tasks, subjects take less time to determine that an item is a word if the presentation of the item is preceded by a presentation of an associated word (for example, "butter" by "bread," or "chips" by "fish"). French proposes to use this priming effect to unmask the computer. He remarks:

> The day before the Test, [the interviewer] selects a set of words (and non-words), runs the lexical decision task on the interviewees and records average recognition times. She then comes to the Test armed with the results ... [and] identifies as the human being the

> candidate whose results more closely resemble the average results produced by her sample population of interviewees. The machine would invariably fail this type of test because there is no a priori way of determining associative strengths ... Virtually the only way a machine could determine, even on average, all of the associative strengths between human concepts is to have experienced the world as the human candidate and the interviewers had. [41, p. 17]

However, French's proposal is illegitimate. The specifications of the Turing test are clear: the judge is allowed only to put questions. There is no provision for employing the equipment necessary to administer the lexical decision task and measure the contestants' reaction times. One might as well allow the judge to use equipment measuring the contestants' magnetic fields or energy dissipation.

The rating games objection is as follows. French claims that ingenious questions such as these can be used to distinguish the computer from the human: on a scale of 0 (completely implausible) to 10 (completely plausible), rate " 'Flugbloggs' as a name Kellogg's would give to a new breakfast cereal" and rate " 'Flugly' as the surname of a glamorous female movie star" [41, p. 18].

French's rating games may be of no assistance at all to the interrogator, however, since the computer is free to attempt to pass itself off as a member of a foreign culture. Conveniently, French claims to discern "an assumption ... tacit in Turing's article" [41, p. 15]: *the computer must pass itself off as a member of the interrogator's own culture.* But French offers no textual evidence in support of this claim. Moreover, he leaves it a mystery why Turing would have wished to impose a restriction that makes the test harder for the computer to pass and yet offers no conceptual gain. French's only source is Turing's 1950 presentation of the test. In the 1952 presentation, Turing says explicitly that the computer is to "be permitted all sorts of tricks so as to appear more man-like" [100, p. 466]. This no doubt includes pretending to belong to a foreign culture.

4.8 Searle's "Chinese Room" Argument

Searle claims that an entity that understands neither the judge's questions nor its own answers could nevertheless "fool native Chinese speakers" and "pass the Turing test" [79, p. 419]. In Searle's thought experiment a human clerk — call him or her "Clerk" — "handworks" a computer program that is capable of passing the Turing test in Chinese. Clerk is a monolingual English speaker who possesses the program in the form of a set of rule-books written in English. Clerk works in a room concealed from the judge's view; she and the judge communicate by passing sheets of paper through a slot. To the judge, the verbal behavior of the Room — i.e., the system that includes the rule-books, Clerk, Clerk's stationery, and the input/output slot — is by hypothesis indistinguishable from that of a native Chinese speaker. In his original presentation — we call it the "vanilla" form — of the argument, Searle claims:

> [Clerk] do[es] not understand a word of the Chinese ... [Clerk] ha[s] inputs and outputs that are indistinguishable from the native Chinese speaker, and [Clerk] can have any formal program you like, but [Clerk] still understand[s] nothing. [A] computer for the same reasons understands nothing ... [W]hatever purely formal principles you put into the computer will not be sufficient for understanding, since a human will be able to follow the formal principles without understanding. [79, p. 418]

There have been numerous objections to Searle's argument. For example, the *systems reply* claims that, while it is true that Clerk does not understand the inputs and outputs, she is merely part of a whole system and the system *does* understand the inputs and outputs. Searle quite properly responds that this reply is worthless, since it "simply begs the question by insisting without argument that the system must understand Chinese" [79, p. 419].

Our (first) reply to the argument is the *logical reply* [21] [29]. It is a straightforward enough point, but seems to have been overlooked by Searle and others. The vanilla form of the argument is not logically valid. The proposition that the symbol-manipulation carried out by Clerk does not enable *Clerk* to understand the Chinese characters by no means entails the different proposition that the symbol-manipulation carried out by Clerk does not enable the *Room* to understand the Chinese characters. (One might as well claim that the statement "The organization of which Clerk is a part has never sold pyjamas in Korea" is entailed by the statement "Clerk has never sold pyjamas in Korea.") In consequence, Searle cannot infer from Clerk's failure to understand the Chinese symbols that a computer performing the same manipulations fails to understand the symbols. (The logical reply is a point about entailment simpliciter and — unlike the systems reply — involves no claim about the truth-value of the statement that the Room understands Chinese.)

Searle has another answer to the systems reply:

> Let the individual ... memoriz[e] the rules in the ledger and the data banks of Chinese symbols, and [do] all the calculations in his head. The individual then incorporates the entire system. ... We can even get rid of the room and suppose he works outdoors. All the same, he understands nothing of the Chinese, and a fortiori neither does the system, because there isn't anything in the system that isn't in him. If he doesn't understand, then there is no way the system could understand, because the system is just a part of him. [79, p. 419]

In this form of the argument, Searle implicitly appeals to a principle that we shall characterize as follows:

> If Clerk (in general, x) does not understand the Chinese input and output (in general, does not Φ), then no part of Clerk (x) understands the Chinese input and output (Φs).

Unfortunately (this is our second reply), Searle does not say why he thinks this "part-of" principle is true. It is conceivable that a homunculus in Clerk's head understands Chinese without Clerk doing so. Likewise for related values of Φ. Conceivably the homunculus produces solutions to certain tensor equations (perhaps this is how we catch cricket balls and other moving objects), even though Clerk herself may sincerely deny that she can solve tensor equations.

A possible response to this reply runs as follows. That Clerk's claim that she is unable to solve tensor equations is sincere does not entail that it is true. The proponent of the part-of principle can simply maintain that, if Clerk catches a cricket ball, then since a part of her is solving tensor equations, so is Clerk. However, this response is not available to Searle. If Clerk's sincere denial that she is able to solve tensor equations is to count for nothing, then consistency demands that one say the same about her denial that she is able to understand Chinese. But it is a cornerstone of Searle's case that Clerk's sincere statement "I don't understand the Chinese input and output" suffices for the truth of "Clerk does not understand the Chinese input and output." This last is the fundamental premiss of both forms of the Chinese Room argument. So Searle is caught on the horns of a dilemma. He can uphold the part-of principle if he abandons first-person incorrigibility; but if he abandons first-person incorrigibility, the whole Chinese Room argument collapses.

Our third reply to the Chinese Room argument is based on the concept of *hypercomputation* [33] (see also Section 5) and is as follows: even if the argument were valid and had true premisses, it still would not show what Searle wants it to show. According to Searle:

> The whole point of the original [i.e., vanilla] example was to argue that ... symbol manipulation by itself couldn't be sufficient for understanding Chinese. [79, p. 419]

> [F]ormal syntax ... does not by itself guarantee the presence of mental contents. I showed this a decade ago in the Chinese room argument. [81, p. 200]

However, the Chinese Room argument cannot show that the symbol-manipulation carried out by a hypercomputer is insufficient for understanding.

A hypercomputer is a machine able to produce the values of functions that are not Turing-machine computable, for example the halting function.

(For some comments on the remarks on hypercomputation in the chapters by Davis and Hodges, see the Postscript.) Turing's 1938 *o*-machines ("oracle" machines) — Turing machines with an (entirely notional) additional "fundamental process" [92] — are a form of hypercomputer. *o*-machines illustrate the fact that the notion of a programmed machine whose activity consists of the manipulation of formal symbols is more general than the notion of the universal Turing machine. (Perhaps the mind, in abstraction from resource constraints and temporal constraints, is a hypercomputer.[74] Searle might find this view as "antibiological" [80, p. 23] as he finds other functionalist views of the mind.)

By its very nature the Chinese Room argument can be applied only against programs that can be simulated by a human rote-worker. The abilities of the rote-worker described by Searle are as stated in the Church-Turing thesis: Clerk does not exceed a Turing machine. In consequence, Clerk cannot simulate a hypercomputer.[75] And so, even if, as Searle claims, there is an implication from

> *x*'s operation is defined purely formally or syntactically

to

> *x*'s operation is neither constitutive of nor sufficient for mind,

this is not an implication that could possibly be established by Searle's Chinese Room argument.

4.9 Turing's Predictions

We conclude that Turing's test survives the standard objections. The question, then, is: will a computer pass the test and, if so, when?

In the 1950 paper Turing predicted that

> in about fifty years' time it will be possible to programme computers ... to make them play the imitation game so well that an average interrogator will not have more than 70 per cent chance of making the right identification after five minutes of questioning. [96, p. 442]

(Turing's prediction is sometimes reported the wrong way round: for example, "Turing clearly believed digital computers could, by the end of the [20th]

[74] Turing's 1936 proof of the existence of functions that are not Turing-machine computable guarantees the existence of an (abstract) model of neuronal function that is not equivalent to any Turing machine: whether or not the brain is equivalent to a Turing machine is an empirical matter (see [21,22,26,72,84,85]).

[75] This argument is discussed further, and an objection rebutted, in [31]; see also [10].

century, succeed in deceiving an interrogator 70 per cent of the time" [106, p. 61].)

In his 1952 radio discussion Turing made a different, overlooked, prediction:

NEWMAN I should like to be there when your match between a man and a machine takes place, and perhaps to try my hand at making up some of the questions. But that will be a long time from now, if the machine is to stand any chance with no questions barred?

TURING Oh yes, at least 100 years, I should say. [100, p. 467]

5 Postscript

Davis and Hodges (see their chapters in this volume) offer a number of objections to the idea of hypercomputation. Since these objections have already been raised and answered in the literature, we shall not discuss them here. Interested readers may turn to e.g. Copeland's [30] (especially Sect. 3 ("The Very Idea of Hypercomputation: Objections and Replies"), where seventeen objections are set out and answered) and the other papers in Copeland's two-volume edition *Hypercomputation* (vols 12.4 and 13.1 of *Minds and Machines*). However, we make the following observations:[76]

Davis asserts, without supporting argument, that quantum computers "can only compute computable functions"; yet this question is in fact open and the subject of much debate (see e.g. the papers in *Hypercomputation* by Calude, Kieu, and Stannett, and [17]).

Davis claims that de Leeuw, Moore, Shannon, and Shapiro [37] show that "Turing machines provided with a random number generator ... could compute only functions that are already computable by ordinary Turing machines"; yet this is not what de Leeuw et al. show or claim to show (as Davis must surely know) — see e.g. their theorem 3 [37, p. 193].

Hodges states that the idea of a random element has "nothing whatever to do with oracles"; yet the concept of a random oracle is well known (see e.g. [3]).

Hodges quotes from a letter by Robin Gandy but fails to include the words (which immediately follow "During this spring he spent some time inventing a new quantum mechanics;"): "it was not intended to be taken very seriously (almost in the 'for amusement only' class)."[77]

Hodges claims that "Turing was probably trying to make quantum mechanics fully predictable"; but Hodges offers no evidence in support of this

[76] We refer to Davis, "The Myth of Hypercomputation," 10 February 2003, and Hodges, "What would Alan Turing have done after 1954?", `http://www.turing.org.uk/philosophy/lausanne1.html`.

[77] Letter from Gandy to Max Newman, undated (but from internal evidence June 1954), in the Modern Archive Centre, King's College, Cambridge (catalogue reference A 8).

claim, merely emphasizing that its supposed truth would rescue his own interpretation of Turing from an important difficulty.

Hodges quotes Church [19, pp. 42–43] and says these remarks show that "Church ... equat[ed] the scope of computability with the scope of machines"; yet the quoted remarks do not show this, for it is evident that Church was discussing (not machines in general but) computing machines (in the words quoted, Church said "it shall be possible to devise a computing machine").

Davis and Hodges both quote Turing's comment "We shall not go any further into the nature of this oracle apart from saying that it cannot be a machine" [92, pp. 172–173], yet neither quotes the immediately following sentence "With the help of the oracle we could form a new kind of machine (call them *o*-machines)" [92, p. 173]. Hodges infers from the first sentence quoted that "oracle machines" are "only *partly mechanical*," but this inference is logically on a par with: "Ink is not a machine, therefore a Turing machine is only partly mechanical." Machines can have parts that are not themselves machines.

Davis objects, "Since all electrons have the same charge, the Copeland-Proudfoot infinite precision real number is actually an integer!"; yet he is aware that we were in fact discussing a *continuously variable* physical quantity, which we called "charge" only "for the sake of vividness" (see further [26, p. 18]).

Hodges asserts (attempting to argue by appeal to authority), "Gandy, as Turing's disciple, never even considered counting an 'oracle' as a kind of machine." However, in an unpublished manuscript Gandy considered at length "analogue machines" based on "physical systems (both classical and quantum mechanical) which when provided with a (continuously variable) computable input will give a non-computable output." (Gandy gave Copeland a copy of this handwritten manuscript shortly before his death in 1995. In it, Gandy claimed that one would not in fact be able to make use of such a physical system to "calculate the values of some number-theoretic non-computable function" but added: "The claim is to be read not so much as a dogmatic assertion, but rather as a challenge.")[78].

References

1. W. Aspray (1990). *John von Neumann and the Origins of Modern Computing.* Cambridge, MA: MIT Press.
2. C. G. Bell, A. Newell (1971). *Computer Structures: Readings and Examples.* New York: McGraw-Hill.
3. C. H. Bennett, J. Gill (1981). *Relative to a Random Oracle* $A, P^A \neq NP^A \neq$ *co-*NP^A *with Probability 1.* SIAM Journal on Computing, 10:96–113.

[78] Research on which this article draws was supported in part by University of Canterbury Research Grant no. U6472 (Copeland) and Marsden Grant no. UOC905 (Copeland and Proudfoot).