The Direct Appeal to Intuition

Alan Turing published “On Computable Numbers" in 1936. Original documents can sometimes possess a greater insight than contemporary texts, especially for new ideas. They are written in a world where no such concept had existed yet, so often the authors will go to a greater extent to justify and explain, rather than just define. Turing’s idea was to simplify the physical process of someone “computing" into an abstract idea, and then reason only about that. An entire section of the original paper was used to convince the reader that his well-defined machine-based definition of "computable" encapsulated the natural idea of "computable". He gave three arguments. An appeal to intuition, a second definition with a proof of equivalence to the machine based one, and numerous examples of numbers that can be computed on Turing machines. Here we annotate his appeal to intuition.

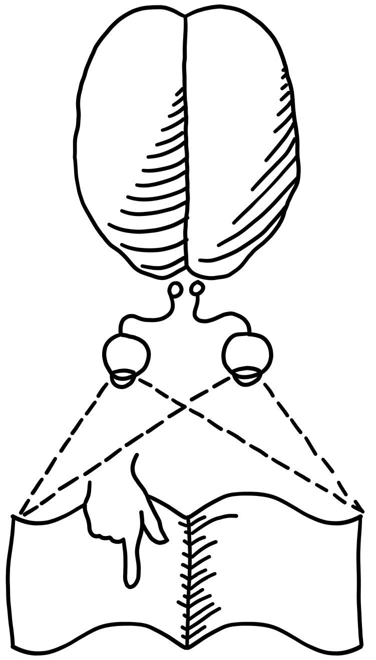

Figure 1: Literal computation being done. A book is edited by a hand, viewed by eyes, and decisions carried out by the brain.

> *Computing is normally done by writing certain symbols on paper. We may suppose this paper is divided into squares like a child’s arithmetic book. In elementary arithmetic the two-dimensional character of the paper is sometimes used. But such a use is always avoidable, and I think that it will be agreed that the two-dimensional character of paper is no essential of computation. I assume then that the computation is carried out on one-dimensional paper, i.e. on a tape divided into squares.*

The first observation made by Turing, is that the physical act of computation is independent of the geometry of this writing surface. You can do the same problems on a book or on a scroll or slate. If you run out of paper, you can always get more. A one dimension infinite tape then is the simplest model to consider. There is no difference between it being unbounded in both directions, or being unbounded in only one.

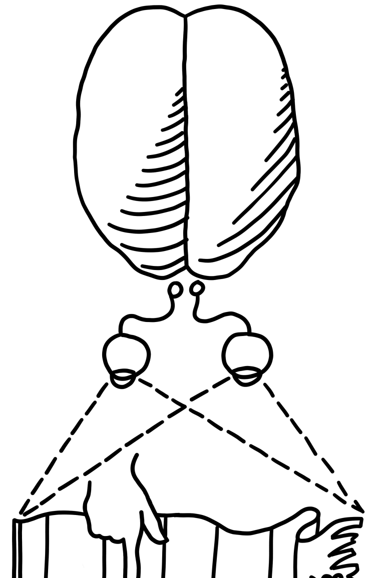

Figure 2: You could order every symbol of the book linearly into a tape

> *I shall also suppose that the number of symbols which may be printed is finite. If we were to allow an infinity of symbols, then there would be symbols differing to an arbitrarily small extent †. The effect of this restriction of the number of symbols is not very serious. It is always possible to use sequences of symbols in the place of single symbols. Thus an Arabic numeral such as 17 or 999999999999999 is normally treated as a single symbol. Similarly in any European language words are treated as single symbols (Chinese, however, attempts to have an enumerable infinity of symbols).*

The second observation by Turing: The size of the tape alphabet has no effect on the possible computation being done. Arithmetic might be easier in one base, versus another, but it can always be done in any base. The properties of this machine that we wish to study are independent of the base size, much like how the primality of an integer is independent of the base it is represented in.

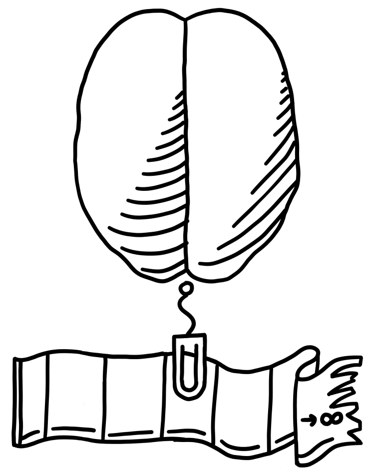

Figure 3: A humble tape head, which can only read and write to a finite amount of cells at a time. In the simplest model, it reads and writes to a single cell

The behaviour of the computer at any moment is determined by the symbols which he is observing, and his "state of mind" at that moment. We may suppose that there is a bound B to the number of symbols or squares which the computer can observe at one moment. If he wishes to observe more, he must use successive observations.

You, as a computer, can really only process information from a finite amount of sources. Eyes are a complex biological input system, but an easy simplification of this is the tape head. It can only read or write to a fixed finite number of cells in a single step.

We will also suppose that the number of states of mind which need be taken into account is finite. The reasons for this are of the same character as those which restrict the number of symbols. If we admitted an infinity of states of mind, some of them will be "arbitrarily close" and will be confused. Again, the restriction is not one which seriously affects computation, since the use of more complicated states of mind can be avoided by writing more symbols on the tape.

Turing states earlier that "the justification lies in the fact that the human memory is nessarily limited." The brain is made up of finite matter, so memory must also be finite. If you also suppose of the existence of a Turing machine with an infinite number of states, then such a machine exists to decide every language. Computing structures are necessarily finite in description.

We may now construct a machine to do the work of this computer. To each state of mind of the computer corresponds an "m-configuration" of the machine. The machine scans B squares corresponding to the B squares observed by the computer. In any move the machine can change a symbol on a scanned square or can change any one of the scanned squares to another square distant not more than L squares from one of the other scanned squares. The move which is done, and the succeeding configuration, are determined by the scanned symbol and the m-configuration.

The entire state of the system is uniquely determined by The current "state of mind", the configuration. The tape head is allowed to move left of right, but only some finite number. This passage also enforces that the machine is deterministic.

Thinking about thinking can be quite difficult, but by distilling computation to its barest essentials, the Turing machine, as a mathematical object, can be reasoned about quite easily. This kind of justification was nessessary to convince a reader, that Turing machines and human computation are the same.

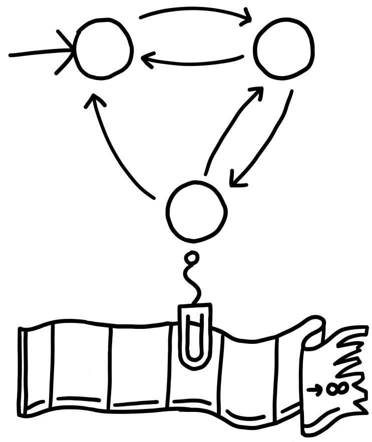

Figure 4: A Turing Machine. A full brain replaced with "finite states of mind"