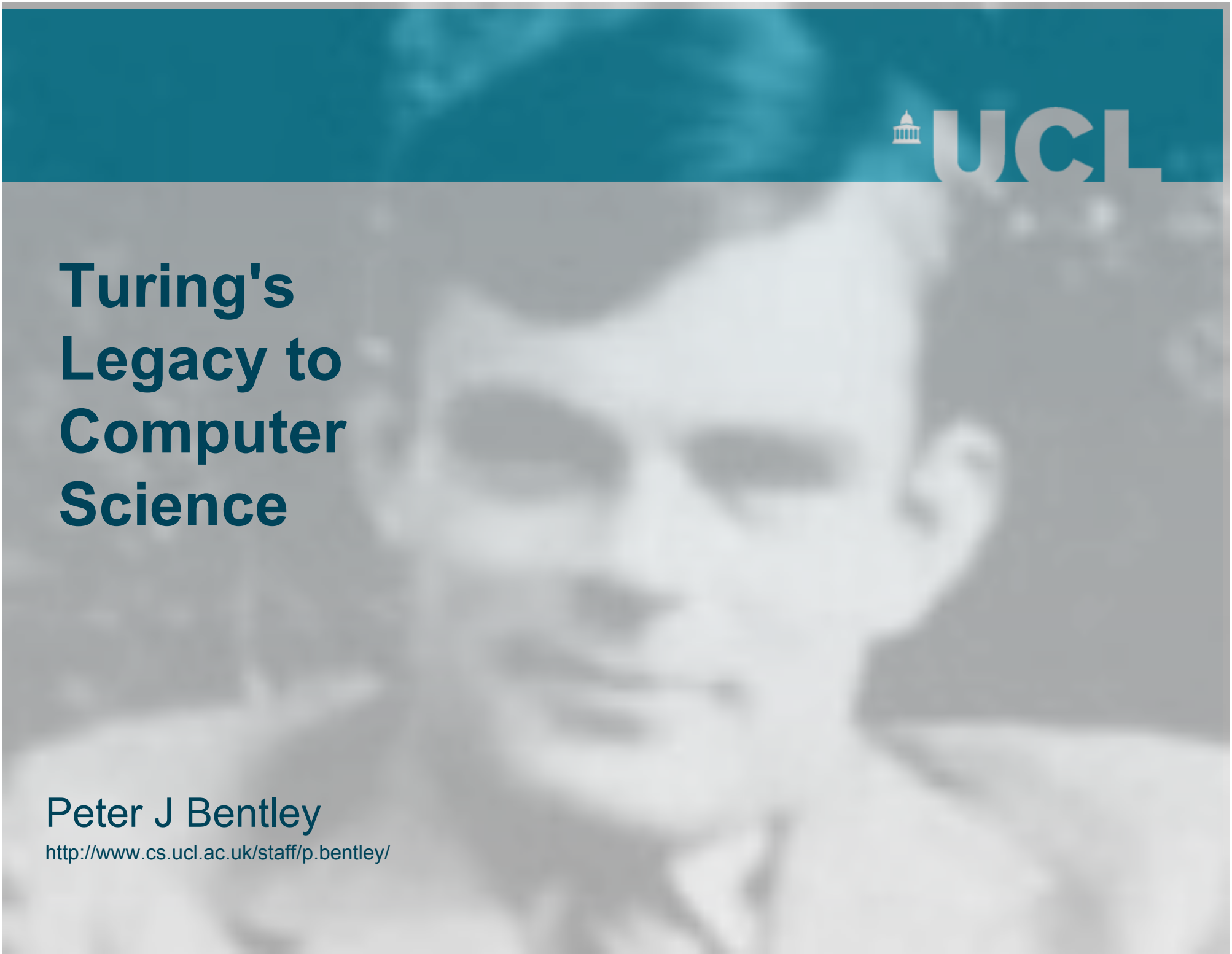


Turing's Legacy to Computer Science

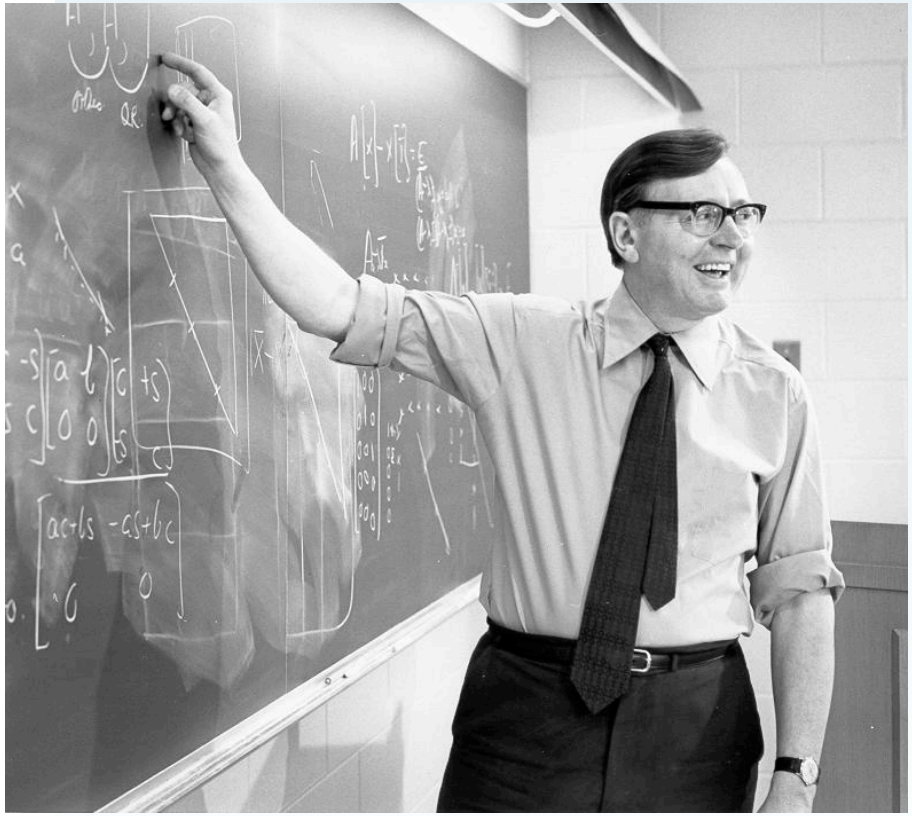
Peter J Bentley
http://www.cs.ucl.ac.uk/staff/p.bentley/

Aged 22, Turing proves the Central Limit Theorem in statistics.

- [Which explains how the distribution of values may resemble a “normal distribution.” For example in a fair coin toss, the histogram of average proportion of heads over frequency will be normally distributed around 0.5. In other words, for most coin tosses, the coin will land heads-up 50 percent of the time or close to 50 percent of the time. Very infrequently will the coin appear to land heads-up less than or more than 50 percent of the time.]

- Great… but someone had already proved it.

It was a sign of his creativity – Turing didn't like to read the work of others before tackling a problem.



colleague James Wilkinson

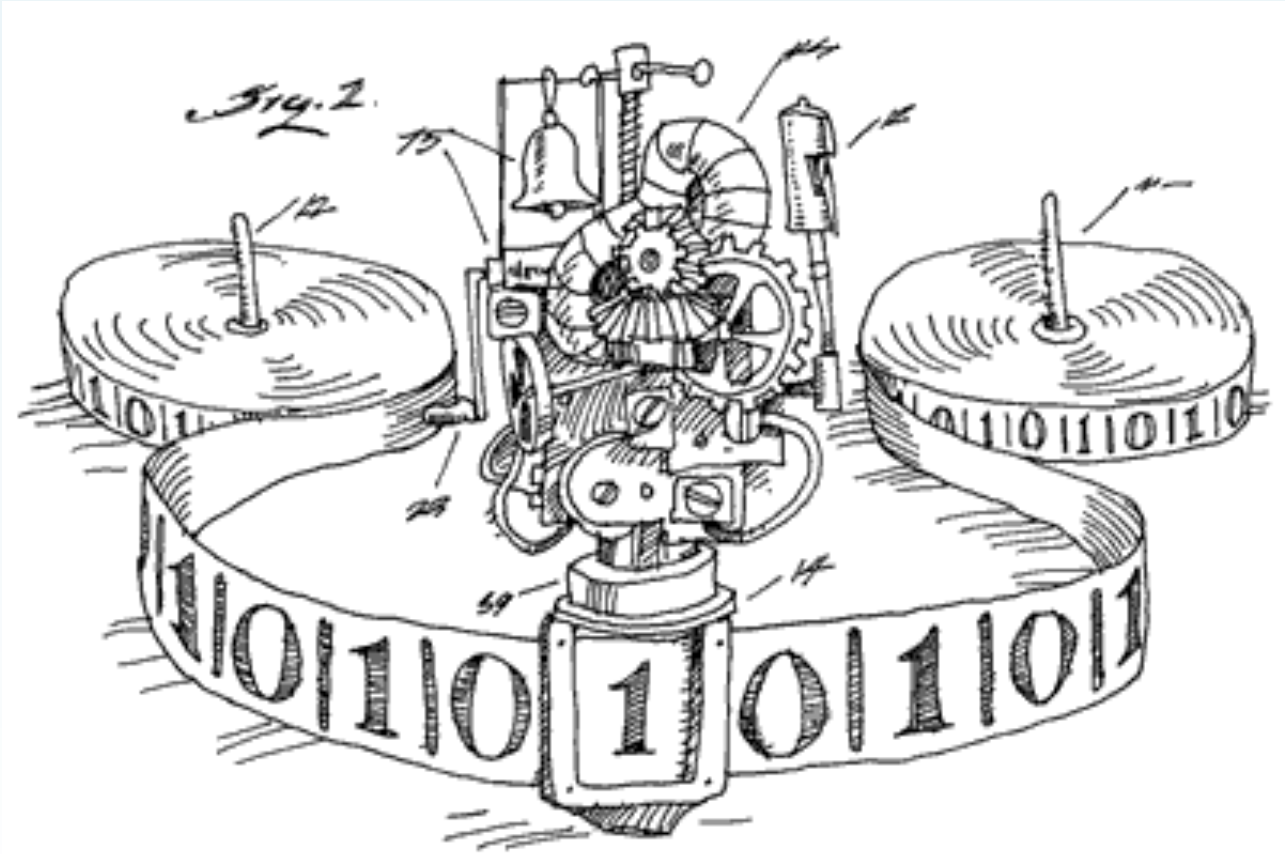
- "Turing had a strong predilection for working things out from first principles, usually in the first instance without consulting any previous work on the subject, and no doubt it was this habit which gave his work that characteristically original flavour.

- I was reminded of a remark which Beethoven is reputed to have made when he was asked if he had heard a certain work of Mozart which was attracting much attention. He replied that he had not, and added:

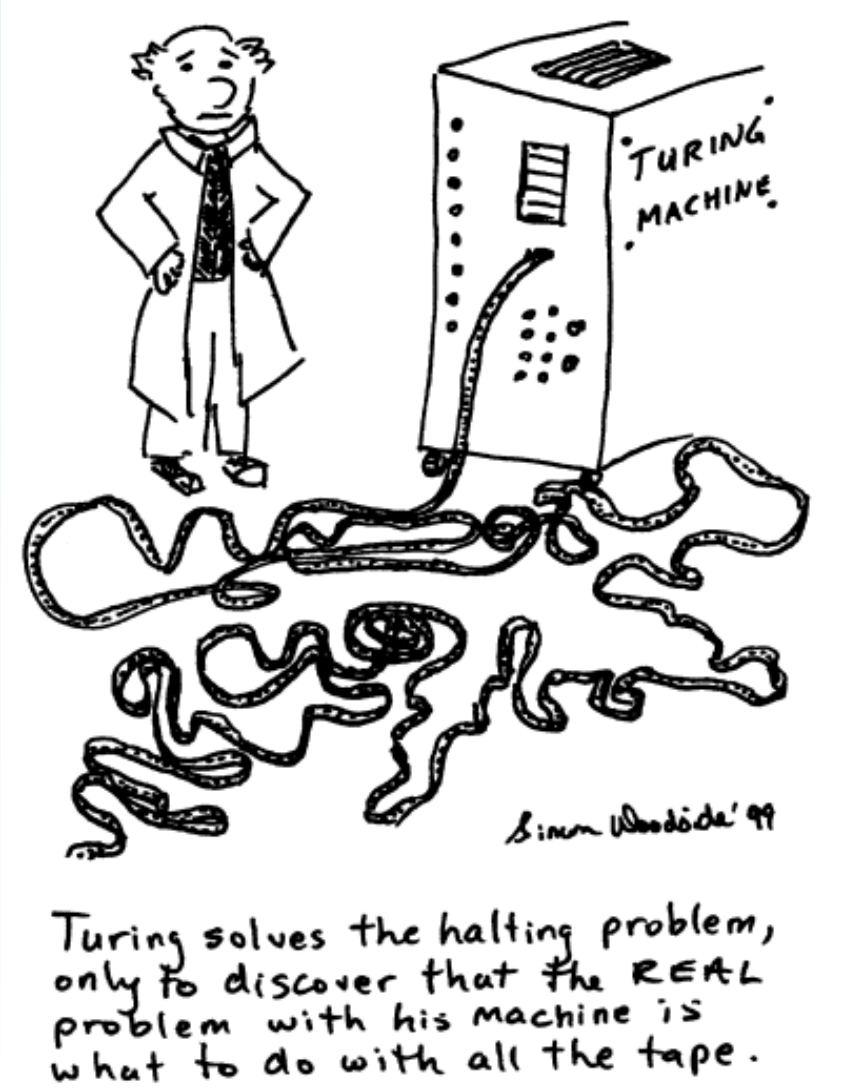
  'neither shall I do so, lest I forfeit some of my own originality.'

Aged 24, Turing solves the *Entscheidungsproblem* challenge posed by a German mathematician David Hilbert, in 1928.

- Is there a procedure that would always tell you if something was true or false, for a given mathematical language ?



The Turing Machine was a Theoretical Computer



Is it possible to tell if this machine will get stuck in an endless loop?

Turing imagined a second machine that would examine the first and halt, outputting “won’t halt” if the first would never halt, or running forever if the first machine did halt.

But what if the second machine looked at itself?

Suddenly there was a paradox: if the machine ran forever then it would stop; but if it stopped then it would run forever. This is logically impossible and so proves that there exist some Turing machines that are undecidable – we will never be able to tell if they halt or not.

But an American called Church had just proved the same thing…

Church-Turing thesis:

Everything algorithmically computable is computable by a Turing Machine

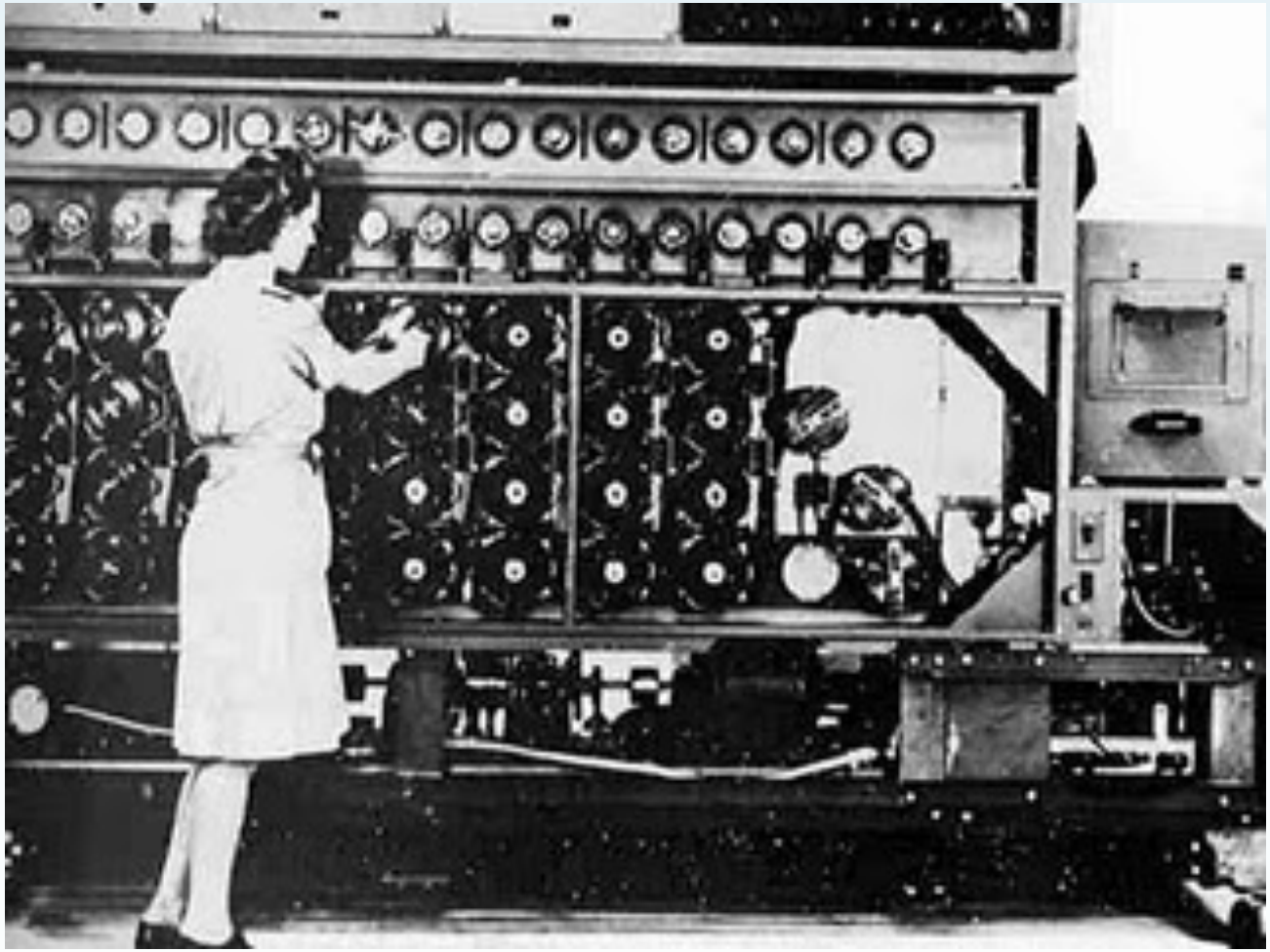
Code Breaking

Turing was invited by the Government Code and Cypher School to help them in their work on breaking the German Enigma codes.

When war was declared in 1939, Turing started working full time at the School in Bletchley Park.

By 1940 had created a machine called the Bombe, which was successfully decoding all German Enigma messages from the Luftwaffe.

By the middle of the next year his work enabled all the Enigma coded messages from the German Navy to be decoded as well.



Code Breaking

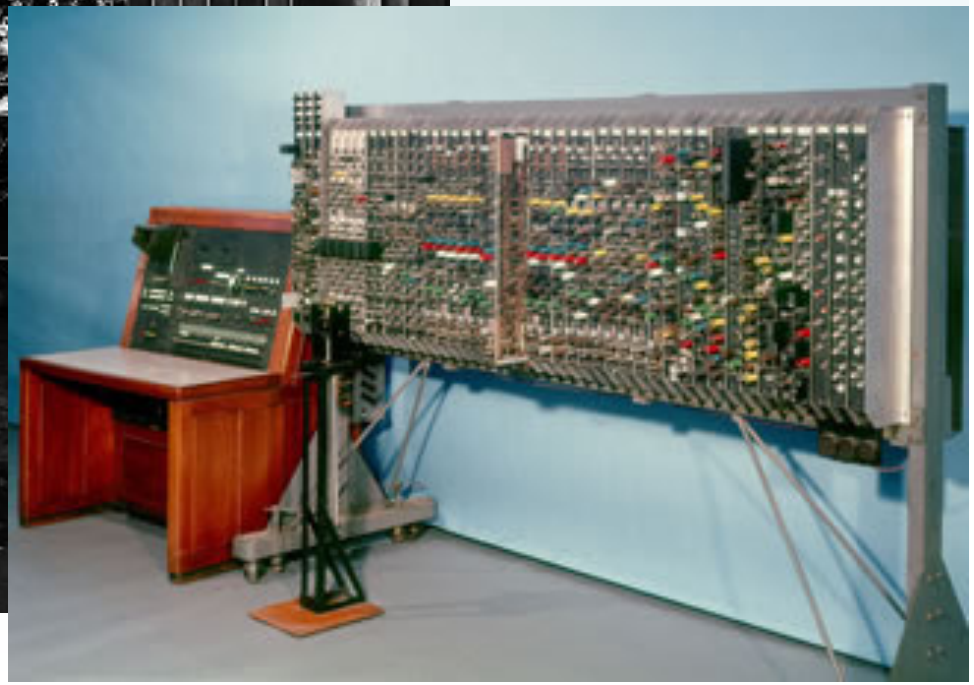
It has been estimated that the work shortened the war by two years.

With around 11 million people a year dying, it was an astonishing achievement.

It is claimed that Winston Churchill said Turing's work was the greatest single contribution to victory in the Second World War.

Automatic Computing Engine (ACE)

After the war, Turing proposed the design for one of the first computers, the Automatic Computing Engine (ACE)



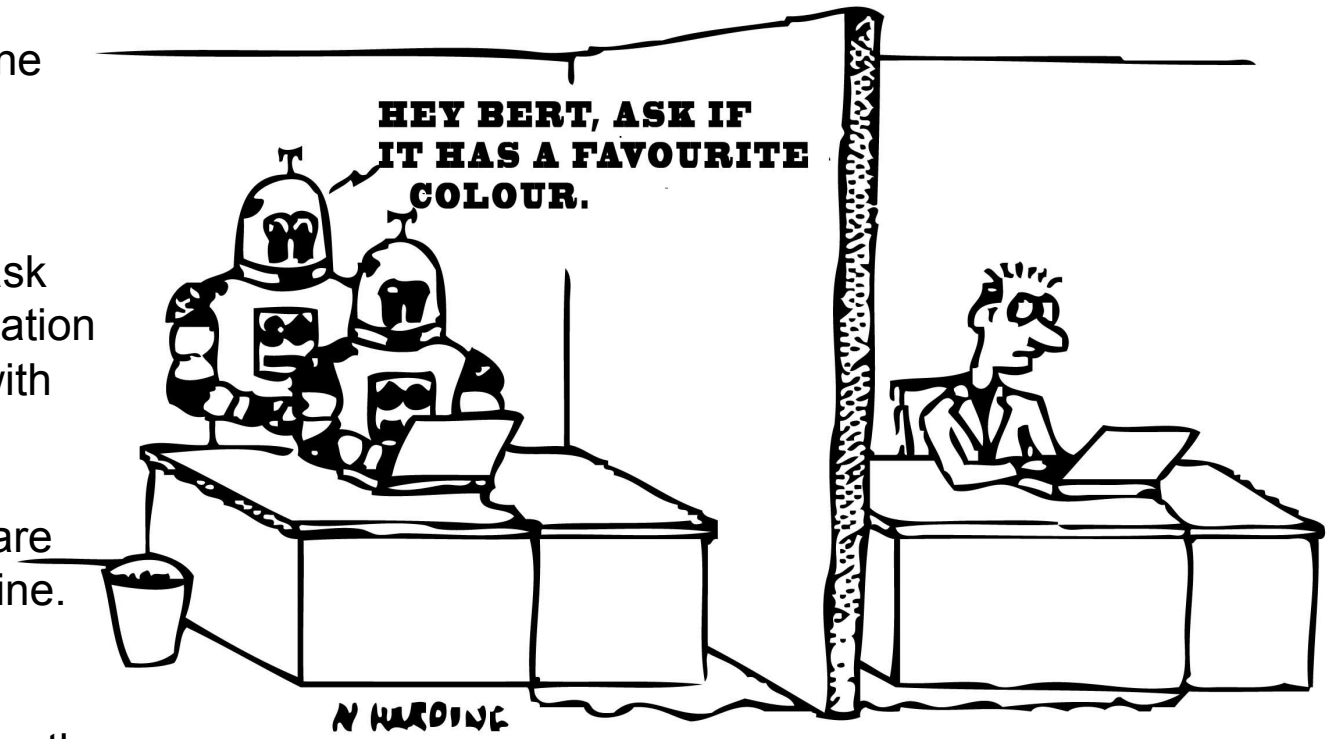
Turing firmly believed that computers would one day be intelligent. He developed the **Turing Test**

There are three rooms, one containing a person, one containing a computer, and one containing you, a human interrogator.

As the interrogator, you can ask questions and hold a conversation (by typing and reading text) with the individual in each room.

You don't know whether you are talking to a person or a machine.

After about five minutes of conversation, if you can't correctly figure out which room contains the computer, then the computer passes the Turing Test.



“I believe that in about fifty years' time it will be possible to programme computers to play the imitation game so well that an average interrogator will not have more than 70 percent chance of making the right identification after five minutes of questioning.”

“I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.”

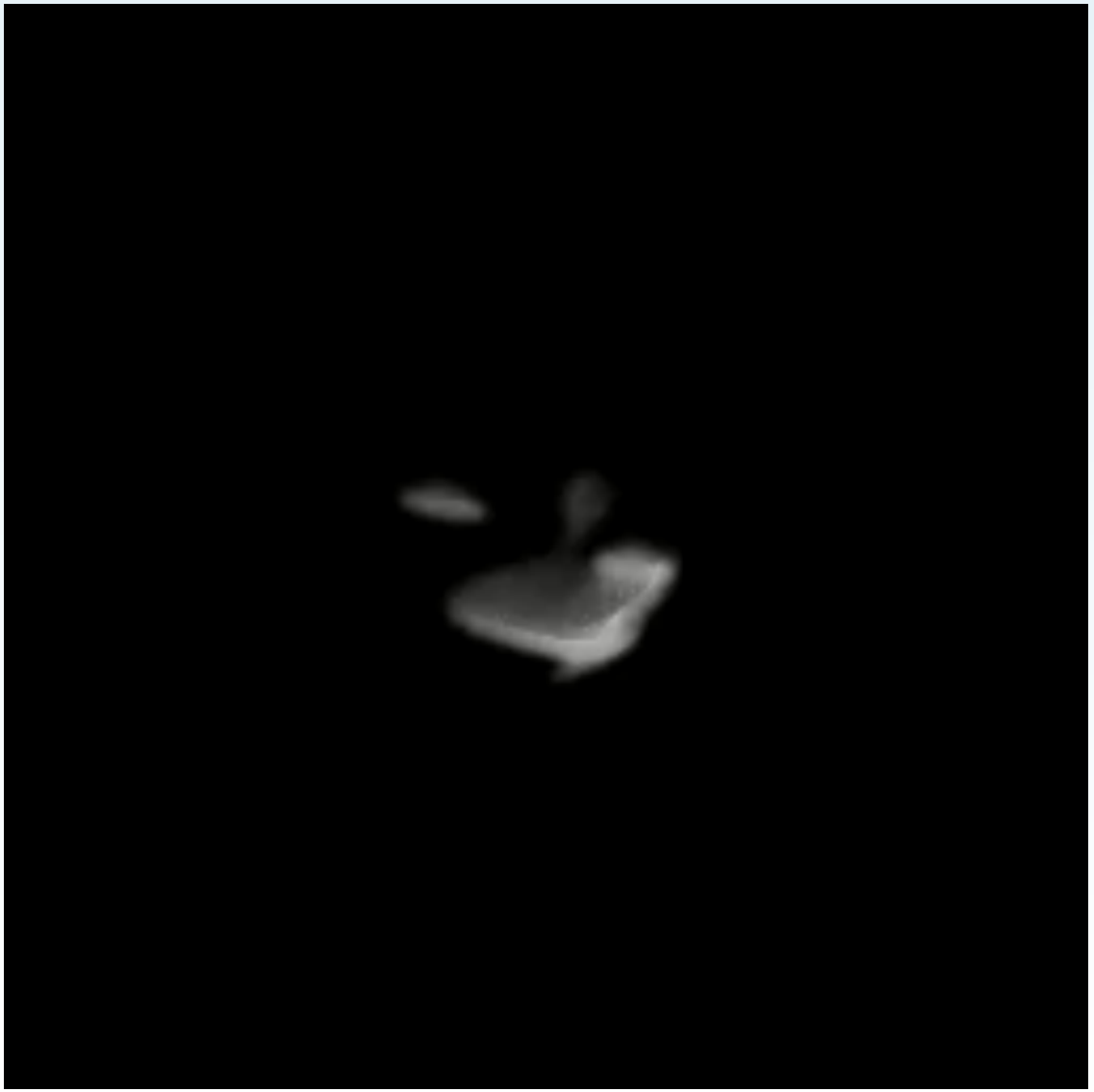

I failed the Turing test

He even looked into Morphogenesis in Biology:

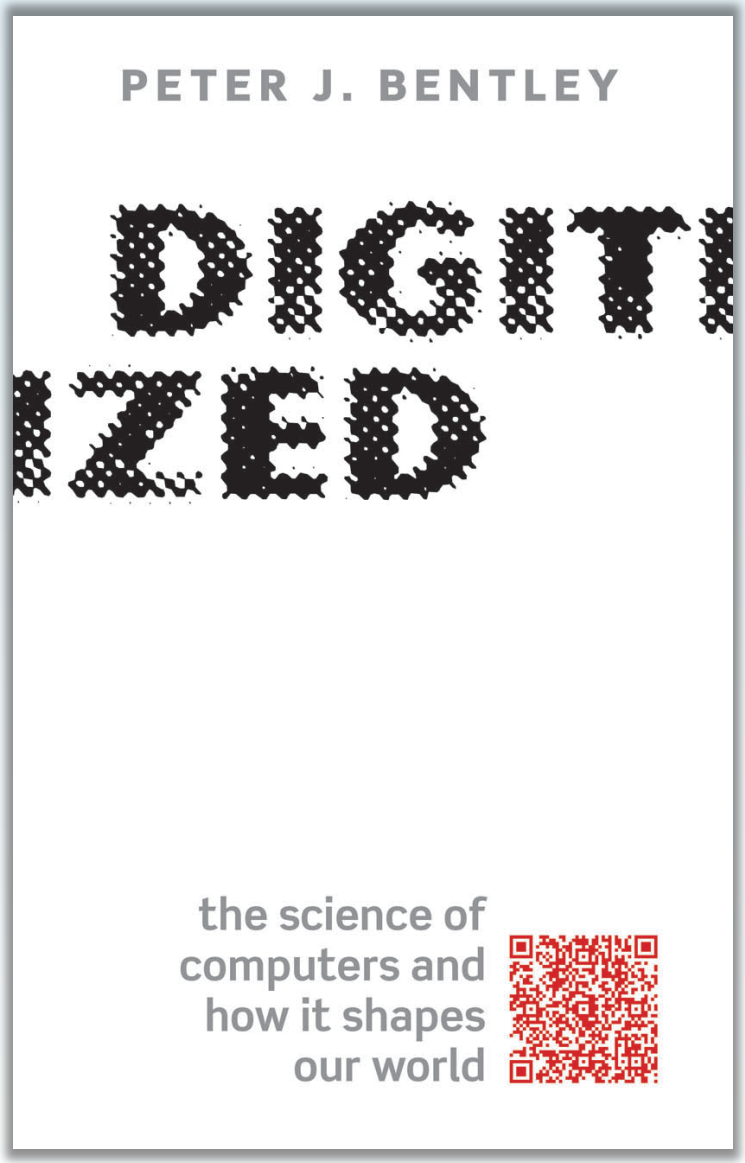
He suggested that the patterns in nature are caused by chemicals that diffuse and react with each other.

The Turing Reaction diffusion equations describe fascinating patterns.

(Morphogenesis in biology does not work like this, but it was a good guess!)




PRIVATE



Bentley, P. J. (2012) ***Digitized***. OXFORD.