Background Notes on Alan Turing: Computing Pioneer, War Hero and Gay Icon

Simon Ritchie

Algorithms

Before the invention of computing machinery, a Computer was a human who carried out computations. Organisations like insurance companies ran teams of computers, sitting at desks and doing whatever calculations the business required – actuarial calculations etc. Human computers were also used to create tables that other people used to do their own calculations.

In the early nineteenth century French mathematicians came up the idea of breaking a computation down into simple steps expressed as an ***algorithm***. (Named after the medieval Arab mathematician al-Khwārizmī.) Since the computers only had to understand how to do each step, they didn't need too much training and were cheaper to hire. It was left to more skilled people to work out the algorithms. Computing was considered a suitable job for young women. Maths graduates, usually men, worked out the algorithms.

An algorithm typically contains loops of repeated instructions and tests, such as:

1. Start with a number
2. perform a calculation on the number
3. take the result
4. if the result is less than 1, stop, otherwise go back to step 2 and repeat the calculation on the result that you just got

The computer follows the algorithm until the resulting number is less than 1 or until it's time to go home. For a given starting number, the algorithm either halts with a result or loops forever.

By the twentieth century, large businesses and government offices employed teams of computers using calculating machines to crank through algorithms.

A modern computer program is an expression of an algorithm.

Cambridge mathematician Charles Babbage promoted the algorithmic approach in Britain. In 1837 he designed (but failed to build) the Analytical Engine, a programmable mechanical computer. The Italian Luigi Menabrea wrote a paper containing a few very simple example algorithms for the machine – to add one number to another etc.

Countess Ada Lovelace (daughter of the poet Lord Byron and resident of Horsley Towers near Guildford) translated the paper into English and added an algorithm that was non-trivial and carried out a real mathematical task – it calculated the numbers of the Bernoulli sequence - making her the first person to write a non-trivial program for a digital computer.

Babbage never managed to build his Analytical engine, but a few years ago, working versions were built for various museums. Babbage's instrument maker predicted that the machine would never work reliably and, sure enough, the working versions are prone to breaking down as parts wear out.

In 1854, during the Crimean War, Babbage figured out how to break the Vigenere cipher. The cipher was used by the Russian Army and thought to be unbreakable. Babbage never published this work and it could well be that the British used it in secret to break Russian communications. As we will see, this pattern is familiar.

Telephony

In the early 20th century, telephony and radio were the hot technologies.

1891 Almon Brown Strowger invented an electromagnetic automatic call router, the Strowger Telephone Exchange.

1904 John Ambrose Fleming invented the Diode Valve and thus created the new field of electronics.

1906 valves used in telephony.

By the 1930's the British telephone network was run by The Post Office (The GPO). The international telephone network was also largely British owned, linking together the greatest empire in the world.

The Post Office had its own training system including sponsoring bright employees to study at University. It ran a research facility at Dollis Hill.

Telephony research produced simple logic mechanisms. Initially these were purely electrical, using devices like relays, but later used electronics.

Much of the research behind modern computing and networking comes out of telephony. By the 1960s the international telephone network potentially allowed any telephone to dial any other telephone, anywhere in the world. Before the Internet was created, it was called "the biggest machine in the world".

Analogue Computers

Before automatic digital computer there were analogue computers. These were mechanical or electro-mechanical devices designed for a specific purpose, such as the Liverpool Tide Predicting Machine. They were used for all sorts of tasks such as controlling production lines or controlling the firing of large guns. A different machine was designed for each task. By the 1970s general-purpose digital computers became cheap and powerful enough to replace them. These days analogue computers are a forgotten technology.

In the 1930s Alan Turing was inspired by the tide predicting machine to design an analogue computer of his own to solve a mathematical problem – finding the first few thousand zeroes of the Riemann Zeta function, in the hope of disproving a conjecture about prime numbers. He never finished this project, it became a casualty of the War. He still had the box of parts when he died.

Proof by Contradiction

Proof by contradiction is commonly used by mathematicians to prove that some conjecture is wrong. To understand Turing's proof of undecidability later, you need to understand this.

An alibi is a very simple example: somebody is accused of committing a crime in Coventry. The accused argues that he can prove that at that time he was in Leicester. To express this formally, to be guilty he would need to be in Coventry and Leicester at the same time, which is a contradiction. Therefore, the initial assertion, that he committed the crime, must be false.

More generally, proof by contradiction works like this:

- Start with some assertion that you want to prove wrong.
- Produce a logical set of deductions from it
- Get to a contradiction

If the steps of the deduction are done correctly, then the initial assertion must be wrong.

You can find some mathematical examples here: http://www.math-cs.gordon.edu/courses/mat231/notes/proof-contradiction.pdf

The Foundations of Mathematics

In 1910 Alfred North Whitehead and Bertrand Russell (then a mathematician, later a philosopher) published Principia Mathematica, an attempt to put mathematics on a rigorous foundation using set theory and symbolic logic. Pages 1-379 of volume 1 led up to a proof that 1 + 1 = 2.

Symbolic Logic allowed mathematical assertions to be expressed formally, and amenable to being manipulated by an algorithm.

In 1928 David Hilbert proposed a programme of research to prove that mathematics:

- is complete: every correctly posed mathematical statement can be proved or disproved
- is consistent (contains no contradictions)
- is decidable: there is an algorithm that can prove or disprove any mathematical assertion that can be expressed in symbolic logic notation

Kurt Godel proved the first two conjectures wrong: formalised mathematics is either incomplete (not every statement can be proved either true or false) or it contains inconsistencies, or both.

This left Hilbert's Entscheidungsproblem (the decision problem): is there an algorithm that can prove or disprove any mathematical assertion?

Hilbert believed that this was so, and started a programme of research to find the algorithm. Other mathematicians were sceptical.

If somebody suggests that something is always true, it's only necessary to find one case where it's not true to prove it wrong.

Mathematicians knew by hard experience that it's easy to design an algorithm that loops endlessly, and ***very*** hard to design one that is guaranteed to halt whatever the input values. Is it possible to create an algorithm A that will check if an algorithm B will halt? This is The Halting Problem. It can be expressed formally, so according to Hilbert, an algorithm must exist that can solve it. Conversely, if it can be proved that such an algorithm can't exist, then Hilbert's entire programme goes up in smoke.

In 1936 Alan Turing published "On Computable Numbers with an application to the Entscheidungsproblem", proving that the Halting Problem, and therefore the decision problem, was impossible to solve. Not bad for an unknown 24-year old.

To do this, Turing proposed a very simple machine that could replace a human computer. The machine would have an unlimited paper tape, divided into squares on which it could write a digit.

The machine worked in binary, so each square could be blank or contain a 1 or a 0. It could move along the tape in steps, square by square, and read the digit from whichever square it was on. It could also erase the square, leaving it blank. It had a simple controller based on the kind of mechanisms used in telephony. As well as the working data, the algorithm that the machine was obeying could be stored on the paper tape.

Turing argued that this machine could do anything that a human computer could do by following an algorithm, and it could mimic any more complicated computing machine, so anything it couldn't do, couldn't be done.

He went on to prove by contradiction that the Halting Problem was not solvable using arguments like:

- Initial assertion: there is an algorithm A that can decide whether any algorithm B will halt when it's run.
- Write an algorithm C that is just like A, except that instead of halting and declaring that B will stop, it enters an endless loop.
- Run algorithm C over itself
- If C will stop, it will run forever, if it runs forever, it will stop – a contradiction

Unfortunately, Professor Alonzo Church of Princeton University published a proof of the same conclusion a couple of weeks ahead of Turing. Church's proof used more conventional mathematics. Turing's argument was much more direct and robust.

Turing went on to show that the systems used in the two arguments were equivalent.

At the time, both results were fairly obscure. There were maybe a few dozen people in the world who understood them. The deeper significance of Turing's paper only emerged much later.

King's College sent Turing to do research at Princeton where he met Einstein, Church and John von Neumann.

von Neumann later worked on the first electronic stored-program computer in 1945 and he bought Turing's ideas to the project.

We now understand that Turing's work is the basis of all modern computing and the underlying theory of computation. All modern computers and all the computers that we can conceive of are limited versions of a Turing Machine. The processor chip is the controller and the limited memory chips replace the unlimited paper tape (which works as long as there is enough memory for the task in hand). The program that the computer has stored in its memory is the algorithm. A typical processor chip is vastly more complicated than Turing's simple controller, but both are equivalent – each can do anything that the other can do.

Babbage's 1837 machine was a mechanical example of the same principle.

Turing was later made a Fellow of the Royal Society for this work.

Turing's interest in automatic computing continued. In the years before the war, he designed an electrical device to multiply two numbers and when the war started he was working on an analogue computer to calculate the zeroes of Riemann's Zeta function, cutting the gears himself. Presumably he hoped to test and disprove the Riemann hypothesis. He still had the box of parts in his house when he died. (The Riemann hypothesis remains one of the great unsolved problems in mathematics.)

It's worth stressing that in those days, most of Turing's peers worked with only pencil and paper. Building a machine to attack a problem in pure mathematics was absolutely not the common approach, in fact it bordered on the revolutionary. As we'll see, Turing's attitude and his eclectic combinations of skills turned out to be very useful later, when he had much more urgent problems to solve.

The Enigma Machine and the Typex

The videos mentioned below present a much clearer picture of the Enigma than I can give in words.

The Enigma machine was originally sold as an encryption device for commercial organisations. In 1928 the British Government Code and Cypher School (GC&CS) bought an Enigma. Hugh Foss analysed it and spotted a flaw. The British produced the Typex, which used the same principles as the Enigma but without the flaw. Cryptanalyst Dillwyn Knox produced paper methods to break Enigma codes.

Knox was a classics scholar, the typical background of a cryptographer in those days. While a Cambridge academic before the First World War he worked on gathering together the fragments of the surviving copy of an ancient Greek poem the Mimes of Herodas, putting them in the right order and guessing at the missing parts. He continued this project on the train to work at the GC&CS and published his translation in 1922.

After the First World War, various British writers including Winston Churchill revealed how successful their colleagues had been at breaking German encryption systems. Bad idea. The German military responded by looking for something better, and bought the rights to the Enigma. They didn't see the flaw but they improved the design anyway to make it yet more secure. By the start of the war, there were several versions of the Enigma in use by the Germans, the Italians and the Spanish, some more secure than others. More improvements emerged as the war went on, and the more secure Enigma designs defeated Knox's methods.

In 1939, Polish cryptanalysts revealed to Knox that they had broken various versions of the Enigma using mathematics, some inspired guesswork and brute-force decryption. They exploited the flaw seen by the British in 1920 and used an electromechanical device called the Bomba to do the hard work. (The bomba was possibly named after the kitchen gadget of the same name - "bombe" in French - a spherical metal mould used to make ice cream.)

(When Germany invaded Poland, the Polish cryptographers escaped to France. Henryk Zygalski later came to Britain and worked at Bletchley Park. After the war he lectured at Surrey University.)

During the early part of the war, the British had never seen a military Enigma machine and they had to guess at its internal workings. Later a few were captured. That often involved remarkable acts of bravery, such as climbing into a scuttled U-Boat and unbolting its Enigma machine during the few minutes before the boat filled with water and plummeted to the bottom of the sea. All done because somebody had said it was important to get hold of that mechanism, but they weren't allowed to say why.

Cribs

The first stage of decryption was to guess part of the original ("plain text") message. Knox's team called that a "crib" (public school slang for notes used to cheat in an exam).

The flaw in the Enigma was that it always transformed a letter into a different letter, never itself. It helped that Enigma machine only had the letters A-Z , no digits or punctuation, so those had to be spelt out. The Germans named army units with Roman numerals, and many numbers expressed that way start with a 1. To make the message clear, they always preceded the number with ROEM ("Roman"), so a lot of German Army messages would contain the text "ROEM EINE". That part of the encrypted result would contain a letter that's **not** R, followed by a letter that's **not** O, and so on. When decrypting Enigma messages, a cryptanalyst becomes very good at spotting things that aren't there.

The same technique works if messages tend to start with a few common sequences – "From General X to Captain Y" and if not that, maybe "To Captain Y from General X". If the first letter of the encrypted version is an F then you know it's not the first version, but it could be the second. If the second letter is an O, neither of those possibilities are true, so you try another guess, and so on.

An example of this appears in the film *The Imitation Game* – one of the few scenes that aren't pure fantasy. During the early days of the war an Enigma operator sent a long message to a colleague. They were testing the radio equipment so the contents of the message didn't matter. He just hit the L key over and over. The British intercepted the message and the analyst immediately noticed that the encrypted version contained every letter except L. The only way that could happen was if the original plain text was all L's. Such a message was easy to break and that would then reveal the contents of all messages on that network for that day.

It proved very easy to find cribs in Italian messages. The Italian Enigma operators sent messages in groups of five letters and added X's to fill the end of the message. That meant that the plain text of almost every message ended with XALTX, XALTXX, XALTXXX or XALTXXXX. Also, most started with with "per" (from) or "destinario" (to) . Bletchley's resulting success at decrypting them allowed the Royal Navy to maintain control of the Mediterranean throughout the war.

Elizebeth and William Friedman

(The non-standard spelling "Elizebeth" is correct.)

The wealthy American industrialist George Fabyan funded and ran his own research institution that investigated subjects that was interested in. He was a Baconian, a follower of the theory that Francis Bacon wrote the works ascribed to Shakespeare. Baconians believed that he had put encoded messages into the texts.

Fabyan assigned two of his research assistants Elizebeth Smith and William Friedman to investigate the encoded messages claim. Elizebeth realised that they would need to use mathematics for that, and they taught themselves the necessary theory, and created a team of cryptanalysts using the techniques that they developed.

Elizebeth and William married in 1917.

When the USA entered the First World War, the government had no cryptography unit and Fabyan volunteered his, run by the Friedmans and growing to 30 people,. They worked through the war, decrypting messages and teaching cryptography to military officers.

During the war, the British planned to use a new cipher machine, the Pletts Cryptograph. To ensure it was secure they invited their allies to try and break it. The Friedmans succeeded, and that project was abandoned. However, the British cryptographers showed no interest in the Friedmans or their mathematical techniques.

After the first war, the Friedmans each worked for different branches of the US government and were not allowed to discuss their work with each other.

William worked for US Army intelligence. In the 1930's the Japanese designed an encryption machine using standard electromagnetic telephony components. Friedman's team called the first version "red" and broke it fairly easily. In 1939, the Japanese produced a more complex machine which the Americans called "purple". Friedman worked on the design of equipment to help break it.

William's team were also early users of modern office automation technology to aid cryptography, using equipment such as IBM punched card collating and sorting machines.

During Prohibition, smugglers delivered alcohol from South America and Canada by boat and used encrypted messages to coordinated the deliveries. Elizebeth Friedman worked for the US Coastguards decrypting the smugglers' messages. When challenged in court to prove that a jumble of letters meant what she said it meant, she called for a blackboard and explained the basics of mathematical cryptanalysis to the jury. The smugglers were convicted. That made headline news in the USA (and alerted the rest of the smugglers to what was going on) but the British continued to use their linguistic techniques.

As well as being a source of alcohol, between the wars South America was a hotbed of German sympathisers and spies. Some of them used the commercial Enigma to send messages. As war with Germany threatened, the US government took an interest and Elizabeth's team were ready with the facilities to intercept their messages, and then to come up with techniques to break them.

The Americans, the British and the Poles all worked on breaking Enigma but were oblivious of each other's progress. Now that cryptology underlies the whole modern economy and is an accepted branch of Computer Science, that seems crazy, but to be fair, things were very different back then.

When the US entered the war, Elizebeth's unit was taken over by Naval Intelligence. A civilian was not allowed to give orders to Navy personnel and a woman was not allowed to give orders to men. She was both, so the unit was run by a man who she had trained a few years earlier.

Before the attack on Pearl Harbour, the Japanese sent an encrypted message to their embassy in Washington to prepare them. It's suggested, with some evidence, that William's team intercepted and broke the message, but couldn't get the news through the system to the right people in time. (That story features in the film *Tora! Tora! Tora!*)

Throughout the second war, Elizabeth worked for the US Navy and William worked for the Army, both doing cryptography. The work took its toll on William, and he suffered a breakdown.

Once the British and the Americans were cooperating on intelligence, there were a number of high-level conferences between the allies, at which Elizebeth was a key figure. She appeared in the attendance records as Mr Friedman, apparently because somebody in the back office assumed that "Mrs Friedman" was a typo.

Mavis Batey's book *Dilly* also mentions William Friedman visiting Bletchley.

After the war, Elizebeth worked for the International Monetary Fund and the World Bank, setting up the secure encryption for the international banking system.

In retirement the Friedmans returned to the Baconian theory and wrote the definitive textbook on the theory that there were encrypted messages in the Shakespeare texts. They concluded that there was no evidence for it, and that the promulgators had cherry-picked the evidence to suit their argument.

Turing's Bombe and Other Work at Bletchley

It's worth pointing out that, although Turing is best known for the work he did at Bletchley Park, he was only there for about two years. He left his mark because much of the work that other people did, followed in his footsteps.

Learning about the bomba, Knox conceived the idea of a more advanced version that could help break the current models of Enigma. Turing designed the solution. Knox later wrote: "At the time of my visit " (to Poland in 1939) "I had ideas which seemed to be better, and I have since discussed them exhaustively with Mr Turing and Commander Travis and we believe that we could produce a really good alarm machine" (memo quoted in Mavis Batey's biography *Dilly*).

Turing had the combination of skills needed to turn Knox's idea into working hardware. Again, the videos give a better explanation than I can here, but in essence, the Bombe solved the most difficult of the cryptographic problems posed by the Enigma. It was an inside-out Enigma machine that, given a crib that an analyst had guessed and the equivalent fragment of an encrypted message, tried the different configurations in order until it found one that transformed the fragment into the crib. At that point it stopped and rang a bell (thus it was an "alarm machine"). It could run through all the possible configurations in about twenty minutes. If it got to the end, the crib was a wrong guess.

Mick Jagger, the producer of the film *Enigma* (yes, that Mick Jagger), paid for a fully working bombe to be built and used as a prop. He donated it to the Bletchley Park Museum and now it's one of their star attractions. They run demonstrations explaining how the decryption process worked from an analyst guessing a crib to the machine cranking through the possible settings.

Enigma wasn't the only encryption device that Bletchley dealt with, and the problem solved by the Bombe was only the most difficult of a whole set. Turing became one of Bletchley's most senior designers, producing clerical procedures, paper devices, electrical and electronic devices to break enemy encryption systems of all sorts. By the end of his time there he was acting as a consultant to all departments and so knew intimate details of all the work they did. This contrasted with the strict security imposed on most of the other thousands of Bletchley workers, who knew exactly enough to do their job and no more. He was also sent to the USA as a consultant to the American codebreakers. He was the codebreakers' codebreaker.

Turing also worked with the Post Office electronics research engineer Tommy Flowers and promoted his idea that valve-based electronics could be used to produce faster decryption systems. After Turing moved on, Flowers went on to design the Colossus, which was used to break the Lorentz cypher machine. This was a much more sophisticated device than the Enigma and carried very sensitive high-level German military traffic, including Hitler's own communications.

(Colossus is sometimes described as the first computer. Actually it had no memory and couldn't be programmed, so not really. The programmable digital computer was invented later in America by a team including John von Neumann, who bought Turing's ideas from 1936 to the project. However, getting to grips with the valve-based logic technology used to create the Colossus certainly gave the engineers who worked at Bletchley a head start after the war, when the British computer industry kicked off. Unfortunately that was compromised by an obsession with secrecy. People who had worked at Bletchley couldn't tell other people what they had invented there.)

The National Museum of Computing (a separate organisation that shares the Bletchley Park site) has a working model of the Colossus, along with lots of examples of machines through the later history of the digital computing industry.

Turing's speciality was the German Navy's version of the Enigma, the most difficult to break. Success with that allowed the British Navy to maintain its dominance of the oceans and eventually to eliminate the U-boat threat. The U-boats were very successful early on at attacking the Atlantic conveys that supplied Britain with food and other materials but they needed fuel and torpedoes. Enigma decrypts allowed the Navy to track the supply ships and the U-boats. They destroyed the supply ships first, and without fuel, the U-boats were easy targets.

Turing wasn't the only mathematician at Bletchley, in fact he wasn't even the first, but he had the right combination of skills to get practical results fast. The rest of the team could have come up with something like the bombe, but maybe not so soon, and time was of the essence. Without Enigma decrypts early in the war, Britain could have been starved or bombed into surrender. The rest of the Empire and later the Americans may then have continued the war in the Pacific but they couldn't have sent troops to invade Italy and France. It would have been left to Russia to invade Europe, and my generation would all speak Russian.

Turing's work alone is said to have shortened the Second World War by several years.

Bletchley wasn't able to break all the Enigma traffic, there was just too much, but for most of the war it was able to break anything that was required. This had to be kept secret. If the Germans had discovered that Enigma had been broken, they could easily have made it less vulnerable. So middle-ranking officers were told that the intelligence came from spies in the German military. They had long experience that information like that couldn't be trusted, so they didn't always act on it. In any case, the whole structure of the armed forces had to be changed to make best use of the information.

Another problem was that simply having too much success would give the game away, so the British had to careful how they used the intelligence.

Winston Churchill had done some intelligence work himself in his youth during the Boer War. He was a great supporter of Bletchley and an avid reader of decrypts. He would embarrass his generals by quoting intelligence that they hadn't bothered to read.

It took until about 1941 to reorganise the armed forces to make full use of the Bletchley decrypts, and even then, there were failures caused by commanders not listening.

The British did run many very successful deception operations during the war: conjuring up fantasy armies, building fake targets to distract bombers, persuading the Germans that the V1s and V2s were missing their target and getting them reconfigured so they did miss their target. Bletchley was central to these, showing whether the deceptions were working and allowing a change of strategy where necessary.

The most famous is probably Operation Mincemeat, the subject of a recent film (and in the 1960's, the basis of the film *The Man Who Never Was*). The invasion of Italy was imminent but the British planted information implying that the target was Greece. Bletchley decrypts gave a running commentary on the success of this stunt, showing the information being analysed and moving up the German Intelligence hierarchy all the way to Hitler, and then troops being ordered to move from Italy to Greece, out of the way of the real invasion.

Remarkably, a similar stunt worked again, when the Germans were persuaded that the Normandy landings were a feint, to draw attention from a bigger landing at Pays de Calais.

The Programmable Digital Computer

The first programmable digital computer ENIAC was produced in 1945 in America by a team that included John von Neumann, who had met Turing in the 1920s and read his paper describing the universal computing machine. Originally intended as a device for producing gunnery tables, ENIAC was used by the Manhattan Project to design the shaped charge that triggered the first atomic bomb. Von Neumann bought Turing's ideas on the universal computer to the ENIAC design and then to its successor EDVAC. He wrote *The Draft Report on the EDVAC*, the first description of a digital computer, so we call the basic pattern of the modern computer "a von Neumann machine", rather than "a Turing machine", although they are essentially the same.

Post-War Work

Turing joined the new National Physical Laboratory and designed the Automatic Computing Equipment (ACE), but the NPL was bureaucratic and progress was glacial.

Post-war American computing design was mostly funded by the US Military and designed for their purposes. The British company Lyons ran a large chain of cafes and they produced the first computer specifically designed for business – Lyons' Electronic Office (LEO). The chief designer went to America to find out about computers, only to find that a lot of the background work had been done in secret in Britain. LEO was used for tasks such as working out the wages and using the weather forecast to predict how much ice cream the cafes would need.

Lyons and Ferranti funded computing research at Manchester University. Turing followed the money and moved there. He continued to be a consultant to the security services.

Turing had always been interested in the idea of machine intelligence and he was one of the early workers in that field. Apart from inventing the computer in the first place, he's known for The Turing Test, based on his Imitation Game. In this game, a person A sits at a keyboard and screen (in those days probably a teletype), exchanging messages with another person B in another room and an intelligent machine C. A can ask B and C questions via the keyboard and see their answers. A's task is to figure out which is the human. If A can't do this, Turing argued that the machine is intelligent.

Call centres now use supposedly artificially-intelligent "chat bots" to answer customer questions. If you've ever been unfortunate enough to interact with one of these, you will know that, seventy years on, Artificial Intelligence research has not produced a device that's anywhere near passing the Turing Test.

Turing's Death

In 1951 British diplomats Guy Burgess and Donald Maclean defected to the Soviet Union. They were both Cambridge graduates, gay and had been spying for the soviets since the 1930s. This caused a huge scandal on both sides of the Atlantic and gay men were suddenly regarded as a security risk.

(As I update this document, the James Webb telescope has just started operating. It's named after one of the early directors of NASA. That choice is controversial because in the 1960's he was responsible for an illegal purge of gay men and women from that organisation.)

In the new atmosphere of paranoia, as a gay Cambridge graduate with deep knowledge of the workings of the security services, Turing was in serious danger. In 1952 he was charged with committing homosexual acts. He pleaded guilty and avoided jail, but had to undergo hormone treatment for a year – chemical castration. He also lost his security clearance and with it, an important part of his identity.

Turing was found dead of cyanide poisoning at his house in Manchester in 1954. He had a cyanide-making experiment running and there was a partly-eaten apple by his bed. The coroner concluded that Turing had deliberately spiked the apple with cyanide. He didn't bother to have the apple tested. Verdict suicide.

Turing's mother and others argued that he had just been careless and got cyanide on his hands. She had often chided him about personal hygiene. Also, his probation and the hormone treatment had ended a year earlier, so why commit suicide then? His death is another enigma.

After his death, Turing remained a relatively obscure figure. Bletchley Park and the work it had done in the war was kept secret until 1975 when information about it was released under the 30-year rule. Turing is now regarded as one of the leading scientists of the 20th Century.

Postscript

- 1954 Turing dies in disgrace
- 1967 Sexual Offences Act legalised homosexual acts - age of consent 21.
- 1975 some information about Bletchley Park declassified.
- 1983 Andrew Hodges' biography of Turing
- Early 1980's the rise of the personal computer. A computer in every house. They were invented by a gay Brit! Who knew?
- 1988 section 28 of the Local Government Act forbids the promotion of homosexuality and the teaching of the acceptability of homosexuality.
- 2000 Section 28 repealed.
- 2001 Alan Turing Memorial, Manchester.
- 2011 Barack Obama list Newton, Darwin and Turing as great British contributors to Science.
- 2013 Turing granted a posthumous royal pardon.
- 2016 Statue of Turing at Surrey University.

Further Reading

- *Alan Turing: The Enigma* by Andrew Hodges pub Hutchinson - the definitive biography of Turing.
- *Dilly: the man who broke Enigmas* by Mavis Batey pub Biteback – biography of Dillwyn Knox.
- *The Code Book* by Simon Singh pub Fourth Estate - covers the whole history of cryptography including Enigma and Turing's work.
- *A Life in Code* by G Stuart Smith pub McFarland & Company – biography of Elizebeth Friedman.
- *Colossus: Bletchley Park's Greatest Secret* by Paul Gannon, pub Atlantic Books - the story of Colossus.
- *Enigma* by Robert Harris pub Hutchinson - Harris' novel describes the working atmosphere at Bletchley. It was made into the film of the same name in 2001.
- *The Imitation game* – a travesty of a film supposedly based on Hodges' book.

Some Useful Videos on YouTube

The Babbage machine:

https://www.youtube.com/watch?v=0anIyVGeWOI

https://www.youtube.com/watch?v=t8aYkow-Fv8

An analogue computer for controlling naval guns: https://www.youtube.com/watch?v=s1i-dnAH9Y4

The Liverpool Tide Predicting Machine: https://www.youtube.com/watch?v=roNyTlGiz5o

The Turing Machine: https://www.youtube.com/watch?v=dNRDvLACg5Q

The Halting Problem: https://www.youtube.com/watch?v=macM_MtS_w4

Enigma:

https://www.youtube.com/watch?v=mcX7iO_XCFA

https://www.youtube.com/watch?v=G2_Q9FoD-oQ

(Both of these videos are a bit confusing at times, but the two together paint a fairly good picture.)

Professor David Brailsford on the Polish code breakers, Turing and Bletchley Park: https://www.youtube.com/watch?v=kj_7Jc1mS9k

Turing's Bombe: https://www.youtube.com/watch?v=V4V2bpZlqx8

John von Neumann: https://www.youtube.com/watch?v=macM_MtS_w4