# 

# 

19 April 2016

**Turing and von Neumann**

Professor Raymond Flood

Thank you for coming to my lecture today. I will be speaking about John von Neumann and Alan Turing.

Both of them had an enormous range of interests not only in pure mathematics but also in practical applications. They made major contributions during the Second World War: Turing on cryptography and von Neumann on weapons development. The Turing machine formalised the idea of an algorithm and the Turing test is important in artificial intelligence while von Neumann founded the subject of game theory. Both are considered founders of computer science.

John von Neumann was born in 1903 in Budapest. Budapest at the turn of the century produced many fine mathematicians and physicists but von Neumann’s brilliance stood out. He was a child prodigy learning several languages and showing early exceptional ability in, and enthusiasm for, mathematics. By age six he could divide two eight digit numbers in his head, by age eight he was studying the calculus and by age twelve he had read and understood Borel’s *Theory of Functions.* He also had an amazing memory and was supposed to be able to memorize at a glance the names, addresses and telephone numbers in a column in a telephone directory. He retained this ability of absolute recall all his life.

Von Neumann’s father was not keen on him becoming a mathematician for financial reasons and they compromised on a career in chemistry. He studied chemistry first of all in Berlin from 1921 to 1923 and then in Zurich for the next two years. Simultaneously he was registered to study mathematics in Budapest although he never attended any lectures there. Nevertheless, in 1926 he received a diploma in chemical engineering from Zurich and a doctorate in mathematics from Budapest!

Von Neumann was famous for his speed of thinking and there are many stories about his ability. One of my favourites is his solution of the famous fly puzzle.

The setup is as follows.Two cyclists start 20 miles apart and cycle towards each other at a constant speed of 10 miles per hour. At the same time a fly starts from the leftmost cyclist and flies towards the rightmost cyclist at 15 miles per hour.

On arrival the fly immediately turns around and flies back until it arrives at the left cyclist when it again turns around and flies to the right.

The fly continues to do this until the cyclists meet. The question is to calculate the total distance that the flies travels.

The slow way to find the answer is to calculate the distance travelled on the first rightwards leg of the fly’s journey then the next leftwards leg then the next rightwards leg etc., etc., and then sum the resulting infinite series of continually decreasing distances.

But the quick way is to notice that the cyclists meet exactly one hour after they start – they are approaching each other at twenty miles an hour and they are 20 miles apart - and the fly is travelling for all of that hour so the answer is 15 miles because the fly is travelling at 15mph!

When von Neumann was asked the puzzle he replied instantly with the right answer of 15 miles. The disappointed questioner said something like: “Oh, you’ve heard it before and know the trick?” Von Neumann replied “What trick – all I did was to sum the infinite series.”

Von Neumann’s doctoral thesis was entitled *The Axiomatic Deduction of General Set Theory*  and was very influential and he kept an interest in set theory and logic for the rest of his life. Towards the end of the 1920s he had academic positions at Berlin and Hamburg.

Two of his main interests during this time were quantum mechanics and operator theory. His basic insight was that the geometry of the vectors in certain infinite dimensional spaces, called Hilbert spaces, has the same formal properties as the structure of the states of a quantum-mechanical system. Von Neumann’s book on Quantum Mechanics appeared in German in 1932 and has been widely translated. The Nobel physics prize winner Eugene Wigner said that Von Neuman’s contributions to quantum mechanics alone “would have secured him a distinguished position in present day theoretical physics.”

In 1928 he published a paper establishing the subject of game theory. Von Neumann’s work on games is characteristic of a lifelong approach of using mathematics in practical situations. The consequences of his work go far beyond its applications to games of chance, such as poker, and have been important in economics, psychology, sociology, politics and military strategy.

A *zero-sum game* between two players is one in which the gain to the winning player exactly equals the loss to the loser, so the total payoff to both players sums to zero. Essentially Von Neumann proved that zero-sum games are not worth playing, in the sense that each player can always find an optimal strategy! Let me explain.

He developed Game theory to cover conflicting situations where:

There may be any (finite) number of players

Each player can take one of a finite number of actions and different players can have a different list of actions they can take.

At each play of the game players do not know what action will be taken by the other players.

The result of the game determines a payment to each player which can be positive, zero or negative.

As I’ve mentioned if the **sum** of payments to all players is zero the game is called a *zero-sum* game and, of course, if there are also two players it is *two person zero-sum* game. A crucial concept in game theory is the *payoff matrix.*

Example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | YOU |  |  |
|  |  | Action α | Action β | Action γ |
| ME | Action 1 | 3 | -5 | -2 |
|  | Action 2 | 6 | 9 | 8 |
|  | Action 3 | -1 | 6 | -3 |

Here we have two players, You and Me. You can take one of three courses of action called α, β or γ while I also have three courses of action 1, 2 or 3. The entries in the payoff matrix give the payment I receive corresponding to our choices of action.

So if I choose action 2 and you action γ I will win £8 and because it is a zero-sum game you will lose £8, in other words you give me £8.

If I choose action 1 and you choose action β the corresponding entry in the payoff matrix is -£5 so you will win £5 and I will lose £5.

So how should each of us play this game with this payoff matrix? The underlying idea for each player’s choice of action is to get the best payoff in a worst case scenario. Another way of putting it is to get the best payoff when your opponent makes his or her best counter move. This is a *risk-averse* strategy. Let us see what it means for the payoff matrix in the example.

I look at the payoffs on taking action 1, over all the actions you could take, and I note the minimum payoff as -5. This is just saying that the minimum entry in the first row is -5 and is the worst that could happen if I take action 1.

Similarly for action 2 the minimum payoff is 6.

And for action 3 the minimum payoff is -3.

Now my risk averse strategy is to *maximise my minimum* payoff which is to take action 2 and obtain a payoff of at least 6.

Let us perform the analogous analysis for you. Remember the payoff matrix is written from my viewpoint. So for the entries written this way you want to *minimise the maximum* number appearing in each column.

So you look at the payoffs on taking action α, over all the actions I could take, and note the maximum entry 6 in the first column which is the worst that could happen to you.

If action β is taken the maximum value in the second column is 9 which is the worst that could happen on taking action β.

If action γ is taken the maximum value in the third column is 8 which is the worst that could happen on taking action γ.

The way to minimize the worst that could happen is to take action α.

Using the risk-averse approach, I want to maximise the row minima and you want to minimise the column maxima. In this case for this game these two numbers coincide with action 2 for me and action α for you, giving a payoff of 6 to me.

Such a combination of actions where the maximum of the row minima coincides with the minimum of the column maxima is called an *equilibrium point* for the game. It is called this because by taking these actions the players guarantee themselves a certain minimum payoff and neither player has a motivation to move from their equilibrium decision. Also, both players can announce their decided course of action in advance to their opponent secure in the knowledge that this information will not allow their opponent to get a better payoff.

This type of equilibrium point is sometimes called a *saddle point*. This is because if we think of my actions as being points, *x*, along the X-axis and your actions as points, *y*, along the Y-axis and the payoff as a function *z* of the pair (*x, y*) then *z* can be thought of as giving a surface. I want to get the highest peak on the surface while you want to get the lowest valley on the surface. So if we have an equilibrium of this sort which is at the same time the highest point in the x direction and the lowest point in the y direction then it looks like the point S in this diagram which is usually called a *saddle point*. It is also called a *minimax point*. The payoff at this point is called the *value* of the game and here is 6.[[1]](#footnote-1)

When a player always takes the same action it is called a *pure strategy.*

So if the game has a saddle point each player has an optimal pure strategy which they can disclose without risking their best possible payoff. But what happens when the game has no saddle point and it is very easy to construct situations where this is the case. This is where von Neumann comes in. He brought in probability and used *mixed strategies*. A mixed strategy is where a player plays each of his actions with a certain probability.

Example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | YOU |  |  |
|  |  | Rock | Paper | Scissors |
| ME | Rock | 0 | -1 | 1 |
|  | Paper | 1 | 0 | -1 |
|  | Scissors | -1 | 1 | 0 |

This is the payoff matrix for the game Rock-Paper-Scissors from my viewpoint playing you. In this game:

Rock beats (smashes) scissors:

Paper beats (covers) rock:

Scissors beats (cuts) paper

This has no saddle point as the maximum of the row minima (-1) does not equal the minimum of the column maxima (+1). Another way of seeing there is no saddle point is that if I announce the action I am going to take, you can always beat me. This could not happen, as we have seen, if there were pure strategies that give a saddle point.

Von Neumann said that the way to play was to use a mixed strategy. One mixed strategy I could use for this game is to choose with probability 1/3 each of Paper, Rock or Scissors. Then we have to view the payoff as the expected payoff calculated using these probabilities or what we would achieve over playing the game many times.

Von Neumann showed, in his paper of 1928, that for every two person zero-sum game there exists a mixed strategy for each player so that the expected payoff for both has the same magnitude when the players use those strategies.

As the game is zero-sum if the payoff is V to one player it is –V to the other and neither can do better than this.

Also each player can reveal their strategy without giving their opponent any advantage.

This result is usually called the *Minimax Theorem* and shows that there exists a strategy or method of playing for each participant that minimises their maximum loss.

Von Neumann later said about this result:

As far as I can see, there could be no theory of games … without that theorem … I thought there was nothing worth publishing until the ‘Minimax Theorem’ was proved.

As I have said von Neumann’s work on games is characteristic of his lifelong approach of using mathematics in practical situations. He published his minimax theorem in 1928 and subsequently collaborated with the economist Oskar Morgenstern.

Their 1944 book *Theory of Games and Economic Behaviour* revolutionised the field of economics.

Here is the title page, and on the right of this slide we see an extract from the book showing the minimax theorem at the bottom. Before von Neumann, mathematical economics tried to imitate the techniques of classical mathematical physics, in particular using analogies between economics and mechanics and used partial differential equations and the calculus of variations.

Game theory has continued to be important in economics. The minimax theorem provides a solution to two person zero-sum games at least theoretically but not so for more general games or more players. In 1994 the Nobel Prize in economics was awarded essentially for a result in pure mathematics. It was awarded to John Nash, John Harsanyi and Reinhard Selten for their work in game theory.

The Nobel citation for John Nash states that he introduced the distinction between cooperative games, in which binding agreements can be made, and non-cooperative games, where binding agreements are not feasible. And that he developed an equilibrium concept for non-cooperative games that now is called Nash equilibrium.

This is an informal statement of Nash’s theorem;

Any n-person non-cooperative game, zero-sum or nonzero-sum, for which each player has a finite number of pure strategies, has at least one equilibrium set of strategies.

Nash’s theorem provides insight into non-cooperative games but there still remain games where participants may agree to cooperate, perhaps in coalitions, and these cooperative games remain as a challenge to be understood. An example is the Prisoner’s Dilemma. *A Beautiful Mind* is the film based on the life of John Nash.

In the late 1920s von Neumann lectured at Berlin and Hamburg. He then taught for three years at Princeton University until he was appointed as one of the founding professors at the newly created Princeton Institute for Advanced Study, a position he held for the rest of his life.

Von Neumann was appointed at the age 30, and was the youngest professor at the Institute, in the School of Mathematics, where he was frequently mistaken for a graduate student. To date 33 Nobel Laureates and 41 (out of 56) Fields Medallists have been affiliated with Princeton Institute for Advanced Study. Another founding professor was Albert Einstein.

By 1936 von Neumann was well established in the Institute for Advanced Study when Alan Turing visited Princeton University. Turing, aged 24, arrived in September 1936 and was to spend most of the years 1937–38 there, leaving after he was awarded his doctorate in June, 1938. During this time he renewed contact with von Neumann whom he had met in Cambridge in 1935.

Let me say a little about Turing’s life up to this point.

Turing was born in Paddington, in June 1912 and went to Sherbourne Boarding School at the age of 13. He was too independently minded to be a conventionally successful student. Instead he studied modern ideas such as Einstein’s Special theory of Relativity on his own.

His grandfather gave him, probably for Christmas 1927, a book on the theory of special relativity – it was an edition of the one shown here - and on the right is the cover of a notebook Alan wrote showing his working and understanding of the book, apparently giving directions to his mother on how to read Einstein’s book.

The notebook has brief comments and guidance on each of the chapters of Einstein’s book. Here are two extracts. The one on the left is the start of the longest comment in the notebook. It is about Chapter XI which is the chapter on the Lorentz transformation and the fifteen year old Turing writes:

Einstein here just gives the equations of the Lorentz Transformation and lets them be taken for granted unless you read his complex proof of Appendix I. I have got a proof though which will give you the result of Chapter XVII directly and if you like the Lorentz Transformation. (Chapter XVII is on Minkowski’s Four-Dimensional space-time).

He then starts his proof which runs for just over three pages and he finishes with the justification:

I have gone rather differently from the book because in this way it should seem less “magicky”

The other excerpt is about the chapter which discusses the equivalence of mass and energy. Turing’s advice to the reader on this chapter is:

*This chapter does not need very careful reading like the others. I should just read it through and think about what interests you.*

On the rest of the page he calculates the energy that would be released by the annihilation of one gram and finds that the energy released would lift one million tons 9 kilometres.

Turing won a scholarship to King’s College, Cambridge and went up in 1931. While an undergraduate one of the prizes he won was a copy of von Neumann’s *Mathematical Foundations of Quantum Mechanics,* which he described as:

*very interesting, and not at all difficult reading, although the applied mathematicians seem to find it rather strong.*

In one of his lecture courses he met the Central Limit Theorem which is a result describing the behaviour of averages.

While still an undergraduate – he was in his third year - Turing provided a rigorous mathematical proof of the Central Limit Theorem. Unfortunately he then found out that it had been proved twelve years earlier! He was nevertheless encouraged to submit the work in support of an application to become a Fellow of King’s. He was successful and at the age of 22 was elected a Fellow of King’s College, Cambridge in 1935.

While waiting for the results of the Fellowship election Turing attended a lecture course on the foundations of mathematics.

In 1928 David Hilbert, during the International Congress of Mathematicians, had posed three questions:

Is mathematics ***complete***, that is, can every statement in the language of number theory be either proved or disproved?

Is it ***consistent***, that is, is mathematics free from contradiction?

Is it ***decidable***, that is, does there exist an algorithm for deciding whether or not a specific mathematical assertion does or does not have a proof?

Kurt **Gödel** had shown in 1931 that the answer to the first question is no - the so-called "first incompleteness theorem". There are statements which can be neither proved nor disproved.

He also proved that if number theory is consistent, then a proof of this fact does not exist using the methods of the first-order predicate calculus - the "second incompleteness theorem".

The last question is called the *decision problem* and was still an open problem when Turing was attending the lecture course.

Turing was intrigued by this problem in mathematical logic.

The decision problem was:

Given a mathematical proposition, can one find an algorithm to decide whether the proposition is true or false?

To tackle this, Turing needed a workable definition of an algorithm, and he identified this with the output of an abstract machine, later called a *Turing machine.* So Turing identified algorithms with Turing machines. Turing’s Thesis was that:

Any “algorithm” can be carried out by one of his machines

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 1 | **0** | 1 |  |  |

This consisted of an infinite tape which acts like the memory in a typical computer. Each square is usually blank initially and can be written with symbols. I am going to illustrate with a three symbol Turing machine which uses the symbols *0*, *1* and blank “ ”. The machine has a head which at any one time is positioned over one square of the tape. Here it is positioned over the leftmost non blank square. With this head, usually called a read/write head the machine can perform three basic operations:

1. Read the symbol on the square under the head.
2. Leave it unchanged or edit the symbol by writing another symbol.
3. Move the tape left or right by one square so that the machine can read and edit the symbol on a neighbouring square.

The final thing we need is to allow a component that could be in any one of a finite number of *states*.

A state tells the machine what to do when each of the symbols is read i.e. what to write and which way to move the tape. Also the machine can change states, depending on the current symbol that was read from the tape.

Here is a Turing machine which starting at the rightmost of a string of ones and zeros performs bit inversion i.e. writes 1 when it reads 0 and write 0 when it reads 1. It has two states, a *stop* state, which does what it says i.e. stops irrespective of what it reads and another state called *state 0*. The table, called a *state table,* gives the instructions for what the machine does for each state and each symbol.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***State*** | **Symbol read** | **Write instruction** | **Move instruction** | ***Next state*** |
| State 0 | Blank | None | Move the tape to the left | Stop state |
|  | 0 | Write 1 | Move the tape to the right | State 0 |
|  | 1 | Write 0 | Move the tape to the right | State 0 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **1** | **0** | **1** |  |  |
|  |  |  | 1 | **0** | **0** |  |  |
|  |  |  | **1** | **1** | **0** |  |  |
|  |  |  | **0** | **1** | **0** |  |  |
|  |  |  | **0** | **1** | **1** |  |  |

The machine now stops.

He then went a step further and envisaged a *universal Turing machine* that could emulate all other Turing machines — the universal machine essentially achieves this by reading both the description of the machine to be emulated as well as the input for it from its own tape. An analogy is the modern computer which can do different tasks if its programming is altered appropriately.

In 1936, in the paper, *Computable Numbers,* shown on the right, Turing answered the decision question in the negative: there are mathematical propositions that are *undecidable* — no algorithm can decide whether they are true or false.

An example that Turing gave was that there was ***no*** algorithm that could decide for any given program and its input whether the program will continue running for ever or will stop. This became known as the *halting problem*.

The idea of a Turing machine became the foundation of the theory of computation.

The resolution of the decision problem was obtained simultaneously by two people: Alan Turing and the American logician Alonzo Church who was based at Princeton. Later in 1936 Turing went to Princeton to study for his doctorate with Church as his supervisor. His thesis was in logic and was titled *Systems of Logic Based on Ordinals.*  At a conference in Princeton in 2012 to mark the centenary of Turing’s birth it was said that Turing’s thesis was one of the two most important ever done at Princeton. The other was by John Nash whom we have already met.

Turing wrote to his mother on arrival at Princeton that:

The mathematics department here comes fully up to expectations. There is a great number of the most distinguished mathematicians here.

He then lists some of them, with von Neumann at the head of the list. The great British mathematician G.H. Hardy was visiting for the term and Turing wrote of him:

At first he was very standoffish or possibly shy. I met him at Maurice Pryce’s rooms the day I arrived, and he didn’t say a word to me. But he is getting much more friendly now.

In 1938 Turing was initially unsure that his Fellowship at King’s would be renewed and encouraged by his father explored the possibility of an academic job in America. There were not many available and there was fierce competition for those that there were, from among others mathematicians fleeing pre-war Europe. Then von Neumann offered Turing a position as his research assistant at the Institute for Advanced Studies. It was a big opportunity but Turing heard that his Fellowship was being renewed and preferred to return to Cambridge. The subsequent careers of von Neumann and Turing were to be strongly influenced by their involvement in the war which was soon to begin.

The mathematician Paul Halmos was a research assistant to von Neumann during the war and wrote:

The year 1940 was just about the half-way point of von Neumann’s scientific life, and his publications show a discontinuous break then.

Till then he was a topflight mathematician who understood physics; after that he was an applied mathematician who remembered his pure work. He became interested in partial differential equations, the principal classical tool of the applications of mathematics to the physical world.

He continued**:**

His papers from this point on are mainly on statistics, shock waves, flow problems, hydrodynamics, aerodynamics, ballistics, problems of detonation, meteorology, and last but not least, two non-classical, new aspects of the applicability of mathematics to the real world: games and computers.

Von Neumann’s contributions to war were manifold. Most often mentioned is his proposal of the implosion method for bringing nuclear fuel to explosion (during World War II) and his espousal of the development of the hydrogen bomb (after the war).

After the war von Neumann led a team at Princeton in the development of a computer, very much based on the ideas behind the Universal Turing Machine. Numbers in the machine were represented in binary form, so that any device holding a digit needed to have only two states.

The machine was completed in 1952. It had 3600 vacuum tubes and a memory of five kilobytes. It was the first stored program computer, unlike previous computers that were programmed by altering their circuitry. Although it had its drawbacks, von Neumann’s design model was very influential in the subsequent development of computers.

Turing’s wartime contributions are probably better known and have recently been the subject of the film *The Imitation Game.* Turing joined the British Government Code and Cypher School which was based at Bletchley Park in September 1939. There he joined the attack on the *Enigma* coding machine.

The Enigma was a German electromechanical rotor-coding machine. In the picture, when a key is pressed, a current flows through the three rotors, is reflected, and then flows back through the rotors to light up a lamp to give the code for the pressed key. Then, crucially, as shown on the right one of the rotors rotates, creating a new pathway for the signals. When the first rotor has made a complete turn, as a result of key presses, the middle one starts to move, and when it makes a complete turn the last one starts to move. So the substitution alphabet is changing all the time.

An additional important complication is that there is a plug board that can be altered to connect letters in pairs so that they were swapped on entering and leaving the rotors.

In each state of the Enigma the substitution alphabet would be a *swapping* of pairs of letters and in particular no letter could be enciphered into itself.

There were 17,576 Rotor settings as each rotor could be inserted in any of 26 positions.

There were 6 ways to put the three rotors in the three positions.

The Plugboard connected seven pairs of letters and could be achieved in 1,305,093,289,500 ways

This gives the total number of keys or setups for the Enigma as

17,576 × 6 × 1,305,093,289,500

Due to espionage, mainly by the French, the internal wiring of the rotors was known. At first Bletchley Park had success in cracking Enigma, that is discovering which setting was being used, but this was highly dependent on the system was being used by the Enigma operators to transmit keys, that is settings for the rotors and plug board, and no longer worked when the system was changed.

Turing job was to try and crack the machine irrespective of the system being used to transmit keys. How this was done is explained very well in both these excellent books. On the left is Simon Singh’s *The Code Book* and on the right Andrew Hodges *Alan Turing: The Enigma.*

Hodge’s book is also an excellent biography of Turing. They both have associated websites: <http://www.turing.org.uk/> and <http://simonsingh.net/cryptography/> <http://www.telegraph.co.uk/history/world-war-two/11151478/Could-you-have-been-a-codebreaker-at-Bletchley-Park.html>

Simon’s book also contains the crossword set in the *The Daily Telegraph* by Bletchley Park to recruit codebreakers. They challenged readers to complete it in less than 12 minutes. Those who did so were invited to a further crossword test and it led to six new recruits. I’ve given a web reference to the crossword that was published in the Daily Telegraph.

Turing and his colleagues analysed the logical structure of Enigma. In particular they used the feature that if the Enigma, in any state, changed A into E then in the same state it would change E into A and this symmetry would hold for all encipherments in a given state. There was a practical advantage to this because it allowed the decipherment operation to be the same as the enciphering. But it was a considerable weakness. Let me indicate how.

A crib is when you know the plain language meaning of a piece of ciphertext. On the slide I show a crib.

Guessed plaintext: w e t t e r

is encrypted into

Known Ciphertext: E T J W P X

Now it was sometimes possible to find *loops* in cribs. There is a loop in this crib:

w goes to e. Then the rotor clicks round once and t goes to w and the rotor clicks round twice more when t goes back to e. Remember in a given state of the machine pairs of letters are swapped.

So now let us have three Enigma machines, the first one at the top in some state S the next one in the middle in state S +1 where the rotor has clicked round and the third at the bottom in state S + 3 where the rotor has clicked round twice more.

We are now going to parallel the loop in the crib by an electric circuit. We will connect the output of each Enigma to the input of the next. So e coming out of the first goes to input e of the second and output t of the second goes to input t of the third and finally output w of the third goes to input w of the first.

But this way of connecting the machine essentially cancels the effect of the plugboards and allowed Turing to ignore billions of plugboard settings. For example the current flowing out of the rotors of the first machine goes through the plugboard from some letter L1 to e but then in the second machine goes back from e to L1. These two plugboards cancel each other out and the same thing happens between the middle and bottom plugboards and the bottom and top plugboards. Turing does not know L1 but there are only 26 letters it could be. We could try a circuit for each of them or have 26 circuits each with a bulb.

I am skipping details but essentially all that needs to happen now is for the rotors of each machine to click round in unison until the circuit is complete and the light comes on! Many more details are in the books by Singh and Hodges.

Both Turing and von Neumann were recognised for their contributions to the war effort with Turing receiving an OBE and von Neumann the Presidential Medal of Freedom.

Apart from their other common interests they also had a shared interest in biology.

Turing had a lifelong interest in the development of pattern and form in living organisms. He looked at how biological systems that are symmetrical at the start can lose that symmetry, and wondered how this could be caused by the dynamics of the way that the chemicals diffuse and react. Using a computer, he carried out pioneering work in modelling these chemical reactions, and published his results in his 1952 paper, *The Chemical Basis of Morphogenesis.*

Underneath is an example of a 'dappled' pattern resulting from one of his systems.

Another major contribution Turing made was his paper on *Computing Machinery and Intelligence* in the journal *Mind.* Turing started:

I propose to consider the question, ‘Can machines think?’

He refined this question by introducing what he called *The Imitation Game.* In this game there is a machine, a woman and an interrogator.

The object of the game, called the *Turing test,* is for the interrogator to determine which of the others is the machine and which is the woman. They can communicate with the interrogator, but only in a manner that gives no clue to their identities. In his previous work with Turing machines, Turing concentrated on what machines cannot do. Now his focus was on what they can do, and in particular whether the behaviour of the brain can be replicated by a computer. The Turing test remains important in philosophy and artificial intelligence.

After the war was over, Turing went to the National Physical Laboratory in London to work on the design of an electronic computer, the Automatic Computing Engine (ACE). His final university position was as Deputy Director of the Manchester Computing Laboratory. Turing continued to be consulted by GCHQ, the successor to Bletchley Park, but lost his security clearance when he was brought to trial for homosexual activities in 1952. He came under intense scrutiny by the intelligence services, who regarded him as a security risk. He died of cyanide poisoning, aged 41, a half-eaten apple beside his bed, and the inquest brought in a verdict of suicide. Von Neumann also died young – in his case at the age of 53. He left unfinished his research program on a new information and computation theory for biological systems, especially the brain. Inspired, I think, by Turing’s universal machine he was developing his theory of self-reproduction of automata five years before the work of Watson and Crick on the structure of DNA and its role in biological self-reproduction.

Von Neumann and Turing made massive contributions to mathematics, computer science and the times in which they lived and their work continues to have an important impact on our world today.

Thank you for coming along and I hope you enjoy next year’s program of lectures.

© Professor Raymond Flood, 2016

1. Source: John L. Casti,  *Five Golden Rules: Great Theories of 20th-Century Mathematics,* Wiley 1996 [↑](#footnote-ref-1)