

Table of Contents

[Chess](#)

[The Essential Turing](#)

[Evaluation Features](#)

[Programming trials](#)

[Turochamp vs. Glennie](#)

[Publications](#)

[Forum Posts](#)

[External Links](#)

[References](#)

[What links here?](#)

[Home](#) * [Engines](#) * **Turochamp**

Turochamp,

a chess program by [Alan Turing](#) and [David Champernowne](#) developed in [1948](#) as chess playing algorithm, implemented as "paper machine". Since there was no machine yet that could execute the instructions, Turing acted as a human CPU requiring more than half an hour per move. One game from 1952 is recorded, which Turochamp lost to one of Turing's colleagues ^[1], [Alick Glennie](#). It won an earlier game versus Champernowne's wife, a beginner at chess ^[2].

Turochamp incorporated important methods of [evaluation](#) ^[3], and also the concepts of [selectivity](#) and [dead position](#), despite it is unclear how this was "implemented" in the game playing experiments.

Champernowne [later said](#) 'they were a bit slapdash about all this and must have made a number of slips since the arithmetic was extremely tedious with pencil and paper' ^[4]. In a [CCC](#) forum post, [Frederic Friedel](#) mentioned a [search depth](#) of up to three [plies](#) ^[5].

Chess

In his 1953 paper 'Chess' in [Bowden's Faster Than Thought](#) ^[6], Turing defines [evaluation features](#), and elaborates on [minimax strategy](#), [variable look-ahead](#), [quiescence](#) and even [learning](#) as an early example of

a [genetic algorithm](#). He does not explicitly mention the name Turochamp, but the 'Machine', and its game versus a human. [Jack Copeland](#) in *The Essential Turing*, 2004 ^[7] on Turing's paper:

Turing says that the system of rules set out in 'Chess' is based on an '[introspective](#) analysis' of his own [thought process](#) when playing (but with 'considerable simplifications'). His system anticipates much that has become standard in chess programming: the use of heuristics to guide the [search](#) through the [tree](#) of possible [moves](#) and counter-moves; the use of [evaluation](#) rules which assign [numerical values](#), indicative of strength or weakness, to [board configurations](#); the minimax strategy; and variable look-ahead whereby, instead of the consequences of every possible move being followed equally far, the 'more profitable moves are considered in greater detail than the less'. Turing also realized the necessity of using 'an entirely different system for the [end-game](#)'.

The [learning](#) procedure that Turing proposed in 'Chess' involves the machine trying out variations in its method of play - e.g. varying the numerical values that are assigned to the various pieces. The machine adopts any variation that leads to more satisfactory results. This procedure is an early example of a [genetic algorithm](#).

The Essential Turing

[Champernowne's](#) quote on Turochamp from *The Essential Turing* ^[8]:

Most of our attention went to deciding which moves were to be followed up. My memory about this is infuriatingly weak, Captures had to be followed up at least to the point where no further captures was immediately possible. Check and forcing moves had to be followed further. We were particularly keen on the idea that whereas certain moves would be scorned as pointless and pursued no further others would be followed quite a long way down certain paths. In the actual experiment I suspect we were a bit slapdash about all this and must have made a number of slips since the arithmetic was extremely tedious with pencil and paper. Our general conclusion was that a computer should be fairly easy to programme to play a game of chess against a beginner and stand a fair chance of winning or least reaching a winning position.

Evaluation Features

[\[9\]](#)

- [Point Values](#) for [Material](#): [Pawn](#)=1, [Knight](#)=3, [Bishop](#)=3.5, [Rook](#)=5, [Queen](#)=10
- [Mobility](#): For the pieces other than Kings and pawns, add the square roots of the number of moves that the piece can make, counting a capture as two moves.
- [Piece safety](#): If a Rook, Bishop, or Knight is defended once, add 1 point; add 1.5 points if it is defended twice.
- King mobility: Use the same method as above, but don't count castling.
- [King safety](#) : Deduct x points for a vulnerable King, with x being the number of moves that a Queen could move if it were on the same square as the one occupied by the King.
- [Castling](#): When evaluating a move, add 1 point if castling is still possible after the move is made. Add another point if castling is immediately possible or if the castling move has just been performed.
- Pawn credit: Score 0.2 points for each square advanced, plus 0.3 points for each pawn defended by one or more non-pawns.
- [Checks](#) and mate threats: Score 1 point for the threat of mate and a half-point for a check.

Programming trials

At the [University of Manchester](#), Turing began programming Turochamp, as well as [Michie's](#) and [Wylie's](#) program [Machiavelli](#), to run on a [Ferranti Mark 1](#) computer, but could not complete them ^[10]. In 2004 [ChessBase](#) published an engine based on Turing's ideas, developed by [Mathias Feist](#) with the help of [Ken Thompson](#) ^{[11] [12] [13]}.

Turochamp vs. Glennie

The mentioned game of Turochamp versus [Alick Glennie](#), who wrote the [Autocode](#) compiler for the [Manchester-Mark I](#) computers. The game took several weeks ^[14].

```
[Event "Paper machine - Human"]
[Site "Manchester, UK"]
[Date "1952"]
[Round "?"]
[White "Turochamp"]
[Black "Alick Glennie"]
[Result "0-1"]
```

```
1.e4 e5 2.Nc3 Nf6 3.d4 Bb4 4.Nf3 d6 5.Bd2 Nc6 6.d5 Nd4 7.h4 Bg4 8.a4 N
xf3+ 9.gxf3 Bh5
10.Bb5+ c6 11.dxc6 0-0 12.cxb7 Rb8 13.Ba6 Qa5 14.Qe2 Nd7 15.Rg1 Nc5 16
.Rg5 Bg6 17.Bb5
```

Nxb7 18.0-0-0 Nc5 19.Bc6 Rfc8 20.Bd5 Bxc3 21.Bxc3 Qxa4 22.Kd2 Ne6 23.Rg4 Nd4 24.Qd3 Nb5 25.Bb3 Qa6 26.Bc4 Bh5 27.Rg3 Qa4 28.Bxb5 Qxb5 29.Qxd6 Rd8 0-1

Publications

- [Alan Turing](#) (1953). *Chess*. part of the collection *Digital Computers Applied to Games*, in [Bertram Vivian Bowden](#) (editor), [Faster Than Thought](#), a symposium on digital computing machines, reprinted 1988 in [Computer Chess Compendium](#), reprinted 2004 in *The Essential Turing*, [google books](#)
- [Alan Turing](#), [Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#)
- [Guy McCrossan Haworth](#) (2013). *Turing, Kasparov and the Future*. [ICGA Journal](#), Vol. 36, No. 1

Forum Posts

- [How did Alan Turing's program work?](#) by Pete Galati, [CCC](#), April 20, 2000
- [New Chessbase Engine called "Turing"](#) by [Ingo Bauer](#), [CCC](#), November 12, 2003 ^[15]
- [Alan Turing's 1950 Chess Computer Program](#) by fourthirty, [Hiarcs Forum](#), September 05, 2017

External Links

- [Classic Computer Chess - ... The programs of yesteryear](#) by [Carey](#), hosted by the [Internet Archive](#) ^[16]
- [Computer und Schach: "Die goldene Gans, die niemals schnattert"](#) by [André Schulz](#), [Spiegel Online](#), November 12, 2003 (German) ^[17]
- [Alan Turing](#) by [André Schulz](#), [ChessBase Nachrichten](#), June 07, 2004 (German)
- [Anmerkungen zur Programmierung der Turing-Engine](#) by [Mathias Feist](#), [ChessBase Spotlights](#) (German, with original article 'Chess' by Alan Turing) (no longer available)
- [An "easy" engine for the Fritz/Rybka interface](#) by [Steve Lopez](#), [USCFSales](#), July 01, 2011
- [The Alan Turing Centenary Conference Manchester UK](#), June 22-25, 2012, hosted by the [University in Manchester](#)
- [Kasparov on Alan Turing and his 'Paper Machine'](#), [ChessBase News](#), June 23, 2012
- [Alan Turing's 60-year-old chess program takes on Garry Kasparov | The Verge](#), by [Bryan Bishop](#), June 26, 2012
- [Kasparov versus Turing](#), June 25, 2012, [Garry Kasparov](#) and [Frederic Friedel](#), [YouTube](#) Video ^[18]
- [In 1950, Alan Turing Created a Chess Computer Program That Prefigured A.I.](#) by [Martin Stezano](#), [History in the Headlines](#), August 29, 2017 ^[19]

References

1. [^ A short history of computer chess](#) from [ChessBase](#)
2. [^](#) Chapter 16, Introduction on 'Chess', in [Alan Turing, Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#), [google books](#)
3. [^](#) [David Champernowne \(1912-2000\)](#), [ICGA Journal](#), Vol. 23, No. 4
4. [^](#) [Alan Turing, Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#), [google books](#)
5. [^](#) [Re: How did Alan Turing's program work?](#) by [Frederic Friedel](#), [CCC](#), April 22, 2000
6. [^](#) [Alan Turing \(1953\)](#). *Chess*. part of the collection *Digital Computers Applied to Games*. in [Bertram Vivian Bowden](#) (editor), [Faster Than Thought](#), a symposium on digital computing machines, reprinted 1988 in [Computer Chess Compendium](#), reprinted 2004 in Chapter 16 of [Alan Turing, Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#), [google books](#)
7. [^](#) [Alan Turing, Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#), [google books](#)
8. [^](#) Chapter 16, Introduction on 'Chess', in [Alan Turing, Jack Copeland](#) (editor) (2004). *The Essential Turing, Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*. [Oxford University Press](#), [amazon](#), [google books](#)
9. [^](#) [An "easy" engine for the Fritz/Rybka interface](#) by [Steve Lopez](#), [USCFSales](#), July 01, 2011
10. [^](#) [Chronology of Computing](#) compiled by [David Singmaster](#)
11. [^](#) [Computer und Schach: "Die goldene Gans, die niemals schnattert"](#) by [André Schulz](#), [Spiegel Online](#), November 12, 2003 (German)
12. [^](#) [Alan Turing](#) by [André Schulz](#), [ChessBase Nachrichten](#), June 07, 2004 (German)
13. [^](#) [An "easy" engine for the Fritz/Rybka interface](#) by [Steve Lopez](#), [USCFSales](#), July 01, 2011
14. [^](#) [Alan Turing vs Alick Glennie \(1952\)](#) from [chessgames.com](#)
15. [^](#) [Computer und Schach: "Die goldene Gans, die niemals schnattert"](#) by [<https://en.chessbase.com/author/andre-schulz>|[André Schulz](#)], [Spiegel Online](#), November 12, 2003 (German)
16. [^](#) [Re: Old programs CHAOS and USC](#) by [Dann Corbit](#), [CCC](#), July 11, 2015
17. [^](#) [New Chessbase Engine called "Turing"](#) by [Ingo Bauer](#), [CCC](#), November 12, 2003
18. [^](#) [Turochamp \(Computer\) vs Garry Kasparov \(2012\)](#) from [chessgames.com](#)
19. [^](#) [Alan Turing's 1950 Chess Computer Program](#) by fourthirty, [Hiarcs Forum](#), September 05, 2017

What links here?

Page

[Alan Turing](#)

[Andre Adrian](#)

[David Champernowne](#)

[Donald Michie](#)

[Engines](#)

[Ferranti Mark 1](#)

Date Edited

Feb 8, 2017

Jan 16, 2012

Mar 29, 2016

Dec 23, 2017

Mar 10, 2018

Jun 2, 2015

Page	Date Edited
Frederic Friedel	Jan 6, 2018
Guy Haworth	Nov 20, 2017
History	Jan 2, 2018
ICGA Journal	Dec 21, 2017
Machiavelli	Jun 2, 2015
Mathematician	Apr 9, 2018
Point Value	Mar 31, 2018
Shaun Wylie	Jun 2, 2015
SOMA	Mar 25, 2015
Steve Lopez	Dec 4, 2016
Turochamp	Dec 14, 2017
University of Manchester	May 24, 2017

[Up one Level](#)