Av Helge Ridderstrøm (førsteamanuensis ved OsloMet – storbyuniversitetet)

Sist oppdatert 15.06.23

Om leksikonet: https://www.litteraturogmedieleksikon.no/gallery/om_leksikonet.pdf

Algoritme

"Algorithms are a step-by-step procedures of formal instructions that solve problems and tasks and accomplish a 'result' in a finite number of steps. Any software or computer operation is the realization of algorithms with the help of a programming language." (Paul 2015 s. 260)

Algoritmer er instruksjoner som følger matematisk-logiske regler. De er basert på matematiske formler kombinert med sorterings- og befalingsstrukturer. Informatikeren Robert Anthony Kowalski lagde denne definisjonen:
Algorime = logikk + kontroll (gjengitt fra Seyfert og Roberge 2017 s. 7).

"*Algorithm* comes to modern English from Arabic by way of Greek, medieval Latin, Old French and Middle English. Historically, it has maintained close ties to the Greek word for number, *arithmós* […] from which the English form *arithmetic* is derived. *Algorithm*'s most common contemporary meaning – a formal process or set of step-by-step procedures, often expressed mathematically […] the word is a 'mangled transliteration' of the surname of a 9th century mathematician, Abū Jafar Muḥammad ibn Mūsā al-Khwārizmī, who lived much of his life in Persia" (Ted Striphas i https://journals.sagepub.com/doi/pdf/10.1177/1367549415577392; lesedato 14.12.20).

Formuleringen av en algoritme må være i et "språk" som en datamaskin kan bearbeide. Programmeringsspråk er et hjelpemiddel for å vise fram og arbeide med på en leselig/visuell måte dataene og kontrollstrukturene i algoritmene (Böck, Ingelmann m.fl. 2017 s. 50).

"A finite sequence of unambiguous steps or instructions designed to solve a complex problem or accomplish a specific task in a way that produces at least one output, for example, a formula used to encrypt data. Algorithms can be expressed in natural language (for example, a culinary recipe or the instructions for assembling an item shipped in pieces), in a symbolic language such as that used in mathematical logic, or in a computer programming language. One measure of proficiency in programming is the ability to create elegant algorithms that achieve the desired result in a minimum number of ingenious steps." (Joan M. Reitz i http://lu.com/odlis/odlis_c.cfm; lesedato 30.08.05)

“In essence, algorithms are simply a series of instructions that are followed, step by step, to do something useful or solve a problem. You could consider a cake recipe an algorithm for making a cake, for example. In computing, algorithms provide computers with a successive guide to completing actions. They’re comprised of a precise list of instructions that outline exactly how to complete a task. [...] Computer algorithms work via input and output. They take the input and apply each step of the algorithm to that information to generate an output. [...] In computing, an algorithm is the list of instructions and rules that a computer needs to do to complete a task.” (https://www.thinkautomation.com/eli5/what-is-an-algorithm-an-in-a-nutshell-explanation/; lesedato 04.11.20)

“Innen informatikken betegner en algoritme en fremgangsmåte for å løse et problem, gjennom en presist formulert instruksjon til en datamaskin. […] Det algoritmene gjør er å ta avgjørelser om *relevans*, dvs. hvilken informasjon som til enhver tid fortjener vår oppmerksomhet. […] Nettopp fordi algoritmer blir brukt til å sortere, rangere og organisere verden for oss, er de langt fra så nøytrale som vi av og til har lett for å tro. Matematikk, systematikk og logikk er ikke unntatt maktrelasjoner. Algoritmene er ikke bare med på å velge *hva* vi skal få vite, men også *hvordan* vi skal få vite om det, og på hvilke premisser. […] algoritmer forstås som forhåndsprogrammerte mekanismer for filtrering og sortering […] Grunnleggende for oppbygningen av en algoritme er betingende setninger av typen: *hvis ... så ... ellers*.” (Taina Bucher i http://tainabucher.com/wp-content/uploads/2009/08/Bucher-algo-frihet.pdf; lesedato 06.02.20)

“An algorithm is said to be *correct* if, for every input instance, it halts with the correct output. We say that a correct algorithm solves the given computational problem. [...] An algorithm can be specified in English, as a computer program, or even as a hardware design. The only requirement is that the specification must provide a precise description of the computational procedure to be followed.” (Cormen, Leierson m.fl. 2009 s. 6)

“Algorithms for event stream processing are being used in the commercial world in order to anticipate intertwined threats and opportunities, such as the propensity of a customer to ‘churn’ and transfer their custom to a new provider.” (Amoore og Piotukh 2016 s. 23) Og såkalte “trending-algoritmer” fanger opp og bearbeider det som er allment populært, særlig på sosiale medier (Tarleton Gillespie i Seyfert og Roberge 2017 s. 78).

I dataprogrammer flettes tallrike algoritmer sammen for å muliggjøre en lang rekke funksjoner. Algoritmer kan skrives i ulike programmeringsspråk. De blir ikke kun skapt av mennesker, men også av andre algoritmer, slik at det oppstår enorme kjeder av algoritmer som gjensidig styrer hverandre (Seyfert og Roberge 2017 s. 25).

"Algorithms are at the core of most technologies used in contemporary computers. Furthermore, with the ever-increasing capacities of computers, we use them to solve larger problems than ever before." (Cormen, Leierson m.fl. 2009 s. 13-14)

"Many useful algorithms are recursive in structure: to solve a given problem, they call themselves recursively one or more times to deal with closely related subproblems. These algorithms typically follow a *divide-and-conquer* approach: they break the problem into several subproblems that are similar to the original problem but smaller in size, solve the subproblems recursively, and then combine these solutions to create a solution to the original problem. The divide-and-conquer paradigm involves three steps at each level of the recursion:

Divide the problem into a number of subproblems that are smaller instances of the same problem.

Conquer the subproblems by solving them recursively. If the subproblem sizes are small enough, however, just solve the subproblems in a straight forward manner.

Combine the solutions to the subproblems into the solution for the original problem." (Cormen, Leierson m.fl. 2009 s. 30)

Sensorbaserte algoritmer er innrettet slik at beslutninger tas når bestemte situasjoner oppstår, og beslutningene tilpasses nye data som kommer inn i systemet (Seyfert og Roberge 2017 s. 190). Et eksempel: "Kontinuerlig måling av glukose i vevsvæske (interstitialvæske) har eksistert siden starten på 2000-tallet, og har revolusjonert diabetes-verdenen. Teknikken har utviklet seg mye, og det har også nøyaktigheten av målingen. Teknikken går ut på at en sensor med en kort fiber settes subkutant i underhudsfettet og sensoren måler innholdet av glukose i vevsvæsken […] På grunn av fysiologien rundt vevsvæsken finnes det ofte en forsinkelse mellom blodets glukoseinnhold og vevsvæskens. Med kontinuerlig måling av sensoren og en avansert algoritme gjøres beregninger for å minske forsinkelsen så langt det er mulig. […] Med denne teknikken slipper personer med diabetes å stikke seg i fingrene like ofte for å kontrollere glukoseinnholdet" (https://www.freestyle.abbott/no-no/my-guide/oppdag-freestyle-libre_2/hva-er-sensorbasert-teknologi.html; lesedato 25.08.21).

"To make a computer do anything, you have to write a computer program. To write a computer program, you have to tell the computer, step by step, exactly what you want it to do. The computer then 'executes' the program, following each step mechanically, to accomplish the end goal. When you are telling the computer what to do, you also get to choose how it's going to do it. That's where computer algorithms come in. The algorithm is the basic technique used to get the job done. [...] Rather than follow only explicitly programmed instructions, some computer algorithms are designed to allow computers to learn on their own (i.e., facilitate machine learning). Uses for machine learning include data mining and pattern recognition." (Stephen F. DeAngelis i https://www.wired.com/insights/2014/09/artificial-intelligence-algorithms-2/; lesedato 04.11.20)

"Algorithms were invented millennia ago to solve simple problems. But in our age of information, they are exerting a sometimes sinister but increasingly vital control over our lives. [...] An algorithm is just a series of instructions – a set of rules to solve a problem. You can perform one in your head or with a pencil. But if there is one thing computers are brilliant at, it's following rules, and the most important algorithms today are all performed by computers, and written in programming language that computers can understand. Type anything into Google and powerful algorithms spring into action to rank each page for you: does it contain the right keywords? How old is it? What else links to it? For predictive policing, they take information on what crime has been committed, where and when, and run it through a sophisticated predictive model and then spit out areas where they think crimes are likely to be committed. The prediction itself is hard to explain, but some have been known to be based on algorithms used to predict the aftershocks of earthquakes. In practice, there might be thousands of such instructions, constantly changing as algorithms learn. The actual machinations of algorithms themselves are closely-held corporate secrets. Look underneath all the code, all the technology and sophistication, and each algorithm, deep down, is just an idea of how to solve a problem. But nothing in an algorithm is above argument or challenge." (Carl Miller i https://www.thenationalnews.com/arts-culture/what-is-an-algorithm-and-how-does-it-work-1.211201; lesedato 04.11.20)

"Love it or hate it, The Rubik's cube is one of the world's most popular puzzles. For many, it is an intimidating challenge, but it doesn't need to be. With a few simple algorithms and some perseverance, you too can solve one in short order if you haven't before. […] The best way to learn how to solve one, without spending hours and days trying to work it out for yourself, is to follow some guides or tutorials. Rest assured this is not cheating. Consider this, a standard 3 by 3 by 3 Rubik's cube has nearly 42 quintillion possible combinations, but only a single correct solution. It would take you a very, very long time to attempt to solve it through brute force alone. You need to know some strategies, technically called algorithms, to short cut the process. In fact, this is the very reason that algorithms exist in the first place. Some of those who have gone through the grinder and learned these algorithms, or set moves, can solve a Rubik's cube in double-quick time. It takes, on average, about 45 minutes, or so, to learn these moves." (https://interestingengineering.com/how-to-solve-a-rubiks-cube-by-using-algorithms; lesedato 11.01.20)

"Sorting is by no means the only computational problem for which algorithms have been developed. [...] Practical applications of algorithms are ubiquitous and include the following examples:

- The Human Genome Project has made great progress toward the goals of identifying all the 100,000 genes in human DNA, determining the sequences of the 3 billion chemical base pairs that make up human DNA, storing this information in databases, and developing tools for data analysis. Each of these steps requires sophisticated algorithms. [...] The savings are in time, both human and machine, and in money, as more information can be extracted from laboratory techniques.

- The Internet enables people all around the world to quickly access and retrieve large amounts of information. With the aid of clever algorithms, sites on the Internet are able to manage and manipulate this large volume of data. Examples of problems that make essential use of algorithms include finding good routes on which the data will travel [...] and using a search engine to quickly find pages on which particular information resides [...].

- Electronic commerce enables goods and services to be negotiated and exchanged electronically, and it depends on the privacy of personal information such as credit card numbers, passwords, and bank statements. The core technologies used in electronic commerce include public-key cryptography and digital signatures [...] which are based on numerical algorithms and number theory.

- Manufacturing and other commercial enterprises often need to allocate scarce resources in the most beneficial way. An oil company may wish to know where to place its wells in order to maximize its expected profit. A political candidate may want to determine where to spend money buying campaign advertising in order to maximize the chances of winning an election. An airline may wish to assign crews to flights in the least expensive way possible, making sure that each flight is covered and that government regulations regarding crew scheduling are met. An Internet service provider may wish to determine where to place additional resources in order to serve its customers more effectively. All of these are examples of problems that can be solved using linear programming [...].” (Cormen, Leierson m.fl. 2009 s. 6-7)

“The Gale-Shapley (GS) algorithm was conceived in a 1962 paper entitled ‘College Admissions and the Stability of Marriage’, in which the writers set out to remove uncertainties in admissions procedures for universities, devising a system that demonstrates no instability in assignment. An assignment is: “optimal if every applicant is at least as well off under it as under any other stable assignment” (Gale and Shapley, 1962: 10). […] the writers look to the model of a community of men and women, in which an even number of members are ranked according to individual preference for marriage (Ibid: 11). Stability is attained through a protocol of repeated rounds of offer-making and rejection; hence the algorithm is known as the deferred acceptance algorithm. [...] Ariely, Hitsch and Hortaçsu (2006) have used data from an online dating service to simulate stable matches between men and women using the GS algorithm, basing their simulations on estimated preference profiles (Ibid: 3). [...] While actual behaviour cannot be described, the GS algorithm can capture some "basic mechanisms in the dating market" (Ibid). Available data includes second-by-second accounts of user activity (Ibid: 6). Match outcomes are simulated using the GS algorithm, and correlations observed in mate attributes (Ibid: 1). The authors note the GS algorithm can also predict sorting patterns in actual marriages, if they exclude the unobservable utility component, search frictions or error terms, such as mistakes made by the user in searching" (Lee Mackinnon i Amoore og Piotukh 2016 s. 219).

"Suppose that you need to hire a new office assistant. Your previous attempts at hiring have been unsuccessful, and you decide to use an employment agency. The employment agency sends you one candidate each day. You interview that person and then decide either to hire that person or not. You must pay the employment agency a small fee to interview an applicant. To actually hire an applicant is more costly, however, since you must fire your current office assistant and pay a substantial hiring fee to the employment agency. You are committed to having, at all times, the best possible person for the job. Therefore, you decide that, after interviewing each applicant, if that applicant is better qualified than the current office assistant, you will fire the current office assistant and hire the new applicant. You are willing to pay the resulting price of this strategy, but you wish to estimate what that price will be. The procedure HIRE-ASSISTANT […] expresses this strategy for hiring in pseudocode. It assumes that the candidates for the office assistant job are numbered 1 through n. The procedure assumes that you are able to, after interviewing candidate i, determine whether candidate i is the best candidate you have seen so far. To initialize, the procedure creates a dummy candidate, numbered 0, who is less qualified than each of the other candidates." (Cormen, Leierson m.fl. 2009 s. 114)

"25. juni 2009, klokken 2:26 lokal tid, begynte ryktet å gå om at popkongen Michael Jackson hadde dødd på UCLA Medical Center i Los Angeles, USA. I løpet av kort tid strømmet millioner av mennesker til Google News for å søke etter informasjon som kunne bekrefte eller avkrefte ryktet. Overraskelsen var stor da de fikk opp denne meldingen i stedet: "Vi beklager, men søket ditt ligner automatiserte forespørsler fra datavirus eller spyware. For å beskytte våre brukere kan vi ikke behandle ditt spørsmål nå." Dette kom som en overraskelse på dem som forventet å få opp ferske og relevante nyhetsartikler. Så hvorfor kom denne beskjeden i stedet? Jo, fordi Google tolket den enorme flommen av søk på de samme ordene "Michael Jackson død" som et dataangrep. Det fikk søkemotorens algoritmer til å reagere slik de var programmert til under fiendtlige angrep: de avviste søkene. Det tok 25 minutter før Google fikk ordnet opp." (Steffen Viken Valvåg og Randi Merete Solhaug https://uit.no/nyheter/artikkel?p_document_id=424614; lesedato 20.10.20)

"In brief, consider the product recommendations one sees on Amazon. These, says the retailer, are the result of one's browsing and purchasing histories, which are correlated with those of Amazon's millions of other customers – a crowd – to determine whose buying patterns are similar to one's own. You, too, might like what this select group has bought, and vice-versa – a process Amazon calls 'collaborative filtering'. Google reportedly works in a similar way. Although the company has moved far beyond its original 'PageRank' algorithm, which measured the number of links incoming to a website to determine its relative importance, it still leverages crowd wisdom to determine what is significant on the web. As *Wired* magazine explained in 2010, "PageRank has been celebrated as instituting a measure of populism into search engines: the democracy of millions of people deciding what to link to on the Web. But Google's engineers … are exploiting another democracy – the hundreds of millions who search on Google, using this huge mass of collected data to bolster its algorithm." (Levy, 2010: 99-100) All this makes algorithmic culture sound as if it were the ultimate achievement of democratic public culture." (Ted Striphas i https://journals.sagepub.com/doi/pdf/10.1177/1367549415577392; lesedato 14.12.20)

Googles Knowledge Vault er en algoritme som skal vise det som er sant i motsetning til det som er mest populært. "Knowledge Vault is a probabilistic knowledge base developed by Google. Based on machine learning, Knowledge Vault is not only capable of extracting data from multiple sources (text, tabular data, page structure, human annotations) but it also infers facts and relationships based on all data available." (https://dejanmarketing.com/knowledge-vault/; lesedato 11.02.21)

I 2006 utlyste filmtilbyderselskapet Netflix en belønning på 1 million dollar til den som kunne forbedre deres anbefalelsesalgoritme med 10 % høyere treffsikkerhet. Algoritmen skulle gjelde den enkelte Netflix-brukers smak og behov, dvs. personliserte anbefalinger. Mange dataingeniører forsøkte å vinne prisen (Seyfert og Roberge 2017 s. 19). "The mission: Make the company's recommendation engine 10% more accurate – or die coding. Word of the competition immediately spread like a virus through comp-sci circles, tech blogs, research communities, and even the mainstream media. ("And if You Liked the Movie, a Netflix Contest May Reward You Handsomely" read the *New York Times* headline.) And while a million dollars created attention, it was the data set – over 100 million ratings of 17,770 movies from 480,189 customers – that had number-crunching nuts salivating. […] CEO Reed Hastings [dvs. Netflix-sjefen] was looking for a way to increase the efficiency of Cinematch, the software the company rolled out in 2000 to recommend movies you might enjoy. […] unlock the secrets of the recommendation algorithm […] Everyone does like movies, and that was undoubtedly in Hastings mind when he first came up with the Netflix Prize. As Gina Keating described in her book *Netflixed: The Epic Battle for America's Eyeballs*, Hastings had lofty ambitions for the competition […] The introduction of streaming video in February of 2007 gave the company more precise information about actual viewer behavior. The recommender system became less about how you rate media and more about what you actually consume. […] Even if the contest didn't play out like many might have first imagined it – one brilliant genius scoring a million dollar jackpot – it instead helped make large strides in the fields of artificial intelligence, machine learning, and recommender systems. Beyond the innovation, this might be one of those cases where the old cliché about how "the real treasure was the friends made along the way" might be true." (Dan Jackson i https://www.thrillist.com/entertainment/nation/the-netflix-prize; lesedato 11.01.21)

"Mye er blitt sagt om hvordan Netflix lager filmer og tv-serier basert på brukerdata. Kreative valg overstyres av maskinlærte estetiske kriterier og hele prosjekter – skuespillere, locations, tematikk – komponeres visstnok av algoritmer." (Nikolai M. Kleivan i *Morgenbladet* 23.–29. oktober 2020 s. 39)

Finansmarkedet opererer med algoritmer som skal forutse utviklinger, med bruk av sannsynlighetsteoremer fra bl.a. lykkespill (Seyfert og Roberge 2017 s. 16). Det er vanlig at algoritmer overføres til nye områder på denne måten.

"Fordi datamaskiner tar over der børsmeglere før rådet grunn, har algoritmehandel (også kalt robothandel) vært kalt finansindustriens svar på den industrielle revolusjon. Algoritmenes store fordel er ikke at de gjør jobben så mye bedre enn mennesker, men de gjør den ekstremt raskt. Spesielt har High Frequency Trading (HFT) fått stor oppmerksomhet. Den sveitsiske børsen har annonsert at den kan tilby en gjennomsnittlig responstid på 34 mikrosekunder. Altså 34 milliondeler av et sekund. Det finnes selvfølgelig ingen mennesker som kunne gjort denne jobben raskere. Kommer da mennesker til å bli overflødige i denne bransjen? […] Å unnslippe menneskelig treghet gjør at aksjehandelen blir mer effektiv, og algoritmene kan sørge for at investorene tjener mer penger. Men mange frykter at dette samtidig kan åpne opp for markedsmanipulasjon. Særlig én hendelse har satt støkk i aksjemarkedet. 6. mai 2010 solgte plutselig en aksjerobot unna teknologiaksjer i et vilt tempo, noe som skapte en kjedereaksjon av robotisert panikk i markedet. Det førte til et lynkrasj på børsen: Dow-indeksen sank som en stein i fem minutter og gikk ned ni prosent, noe som er ny tapsrekord i løpet av én dag. Aksjeroboten skapte nesten en ny finanskrise helt på egen hånd fordi ingen andre lenger hadde noen kontroll. […] Årsaken var feilprogrammering, men nylig ble likevel en mann siktet for å ha startet det hele. Jeg tror ikke en slik siktelse er veien å gå hvis man vil hindre at det skjer igjen. Da kan man ikke straffe dem som avslører feil og svakheter i robothandelen" (Steffen Viken Valvåg og Randi Merete Solhaug https://uit.no/nyheter/artikkel?p_document_id=424614; lesedato 20.10.20).

Den 15. januar 1990 sluttet det nordamerikanske telefonnettet AT&T plutselig å virke og var ute av funksjon i ni timer. Halvparten av alle utenlandske samtaler var umulige å gjennomføre, som førte til et tap på 60 millioner US dollar, samt senere økonomiske tap for AT&T. I ettertid ble nedetiden kalt "The AT&T Crash" og brukt som materiale i Bruce Sperlings sakprosabok *The Hacker Crackdown* (1992) (Shintaro Miyazaki i Seyfert og Roberge 2017 s. 179). Grunnen til sammenbruddet var en liten programmeringsfeil, som ikke ga seg utslag mellom installeringen av ny programvare i desember 1989 og 15. januar året etter. Ingen vet helt hvorfor.

“Innen forskning på kunstig intelligens (KI) står algoritmer sentralt. Genetiske algoritmer har for eksempel som mål å etterape den naturlige evolusjonsprosessen. Ved hjelp av naturlig seleksjon leter man ganske enkelt etter den beste algoritmen. For eksempel den som er best egnet for søk […] Man kan la to algoritmer konkurrere mot hverandre. For eksempel kan den ene være programmert til å gjemme seg, den andre er programmert til å søke etter den som gjemmer seg. Den som vinner av de to, sparer man. Deretter lar man denne konkurrere mot en annen algoritme. Den beste av de to sparer man. Så gjør du det samme mange ganger til. Helt til du sitter igjen med den algoritmen som klarte seg best av alle. Altså den som er best tilpasset oppgaven […] Allerede på slutten av 1980-tallet lanserte General Electrics verdens første produkt laget med genetisk algoritme. Det var et stormaskinbasert verktøysett utviklet for industrielle prosesser. I dag blir genetiske algoritmer brukt innenfor flere felt, blant annet informatikk, kjemi, økonomi og industriteknikk.” (Steffen Viken Valvåg og Randi Merete Solhaug i https://uit.no/nyheter/artikkel?p_document_id=424614; lesedato 20.10.20)

“En økende mengde journalistiske artikler blir nå skrevet ved hjelp av algoritmer, og det er ikke lett for leserne å se forskjell på disse og artikler skrevet av mennesker. Ifølge en studie ved Karlstad Universitet i Sverige klarte i alle fall ikke leserne å se noen forskjell på to sportsartikler, hvorav den ene var skrevet av et menneske og den andre av en datamaskin. Men det finnes også måter å bruke algoritmer på som er langt mer alvorlig. I krigføring og i kampen mot terrorister benyttes de til å finne ut hvilke terrorister som er farligst. Deretter lar man datastyrte droner drepe dem. West Point code er et eksempel på en slik algoritme. Problemet er åpenbart – man vet ikke bestandig 100 prosent sikkert hvem man dreper. […] Sikkerhetskontroller på flyplasser er også delvis algoritmestyrt. Før du kommer dit kan du ha lagt igjen en god del digital informasjon i databaser som kan overvåkes av tredjepart. Navnet ditt, hvilket land du kommer fra, religionen din, hvilken type mat du bestiller på flyet – alt dette kan gjøre at du havner i søkelyset. Det kan også føre til at du blir tatt til side i sikkerhetskontrollen og undersøkt ekstra nøye.” (Steffen Viken Valvåg og Randi Merete Solhaug i https://uit.no/nyheter/artikkel?p_document_id=424614; lesedato 20.10.20)

Betegnelsen “algorithmic culture” “refer to the ways in which computers, running complex mathematical formulae, engage in what’s often considered to be the traditional work of culture: the sorting, classifying, and hierarchizing of people, places, objects, and ideas. […] algorithmic culture then feeds back to produce new habits of thought, conduct, and expression that likely wouldn’t exist in its absence – a culture of algorithms, as it were. The worry here, pointed out by Eli Pariser and others, is that this culture tends to reinforce more than it challenges one’s existing preferences or ways of doing things. This is what is often called “personalization,” though Pariser calls it a “you loop” instead. By the same token, it is possible for algorithmic systems to introduce you to cultural goods that you might not have encountered otherwise. Today, culture may only be as good as its algorithms.” (Ted

Striphas i https://medium.com/futurists-views/algorithmic-culture-culture-now-has-two-audiences-people-and-machines-2bdaa404f643; lesedato 05.06.20)

"Chess is a two-player strategy board game played on checkered board with 64 squares arranged in an 8x8 grid […] chess is considered as abstract strategy game […] Actually, chess bot works like any other computer works, which is by reducing the problem to a bunch of dumb calculations. No worries! the essence of chess algorithms are simple, although, the modern chess algorithms itself kind of complicated. […] applies Minimax Algorithm. Minimax is a decision rule used in artificial intelligence, decision theory, game theory, statistics, and philosophy for minimizing the possible loss for a worst case (maximum loss) scenario. When dealing with gains, it is referred to as "maximin" to maximize the minimum gain." (Fernaldi Fauzie i https://medium.com/analytics-vidhya/how-chess-algorithm-works-69e8ae165323; lesedato 25.08.21)

"Chess was a good fit for computers:
- Clearly defined rules
- Game of complete information
- Easy to evaluate (judge) positions
- Search tree is not too small or too big

1950: Programming a Computer for Playing Chess (Claude Shannon)

1951: First chess playing program (on paper) (Alan Turing)

1958: First computer program that can play a complete chess game

1981: Cray Blitz wins a tournament in Mississippi and achieves master rating

1989: Deep Thought loses 0-2 against World Champion Garry Kasparov

1996: Deep Blue wins a game against Kasparov, but loses match 2-4

1997: Upgraded Dee Blue wins 3.5-2.5 against Kasparov

2005: Hydra destroys GM Michael Adams 5.5-0.5

2006: World Champion Vladimir Kramnik loses 2-4 against Deep Fritz (PC chess engine)

2014: Magnus Carlsen launches "Play Magnus" app on iOS where anyone can play against a chess engine that emulates the World Champion's play at 21 different ages (5 to 25 years)."

(Rune Djurhuus i https://www.uio.no/studier/emner/matnat/ifi/INF4130/h17/undervisningsmateriale/chess-algorithms-theory-and-practice_ver2017.pdf; lesedato 25.08.21)

Den algoritmisk styrte børshandelselskapet Knight Capital tapte i 2014 over 400 millioner dollar på 45 minutter på grunn av en forstyrrelser i en handelsalgoritme (Seyfert og Roberge 2017 s. 26). “Back in 2014, a company called Knight Capital went bankrupt after losing some $440 million in 45 minutes and disrupting the stock market. […] It all started with some technical debt, 8 years old legacy code that hadn’t been used since God knows when, and an unstable delivery procedure, new automated trading code made it on to 7 out of 8 SMARS servers. However, the eighth server still had that legacy code. And the time came when a repurposed configuration value activated the eighth server, and it began making automated trades at such speed as if the Flash was operating it, causing a disruption in the prices of hundreds of stocks and moving millions of shares.” (https://jaxenter.com/top-4-devops-scary-stories-151443.html; lesedato 11.01.21)