

Kodering

“Is jy wakker? Jy moet gou eers jou ontbyt eet, daarna jou tande borsel en dan aantrek. Onthou ook om te kyk of jou tas gepak is en sit jou kosblik en leesboek in jou tas.” Klink dit bekend? Net soos wat jou ouers en onderwysers vir jou vertel wat om op sekere tye te doen, moet ons ook vir **rekenaars** vertel wat om te doen en hóé om iets te doen.

Elektroniese toestelle, soos [selfone](#), skootrekenaars en tablette, het kodes nodig om reg te werk. Selfs moderne tegnologie soos verkeersligte, slim-TV's en motors het interne stelsels wat gekodeer word.



Kodering word ook rekenaarprogrammering genoem en dit is die manier hoe mense met rekenaars kommunikeer. ¹ Kodes word geskryf en hierdie kodes is as't ware **instruksies** wat rekenaars gebruik om take te voltooi. Kodering word onder meer gebruik om webwerwe te bou, apps te skryf en om rekenaarspeletjies te ontwerp.

Hoe werk kodering?

As jy in Frankryk is en jy kan nie Frans praat en 'n taxibestuurder kan nie Engels praat nie, het jy 'n vertaler nodig om vir hom te verduidelik waarheen jy wil gaan. Dieselfde geld vir rekenaars. Hulle verstaan nie ons taal nie en kodering vertaal ons taal vir rekenaars.

Kodes word in syfers en letters in reekse geplaas wat die rekenaar kan verstaan. Sodra daardie boodskappe ontvang word, kan die rekenaar sekere take doen soos hoe om die lettergrootte en -kleur van die teks te verander of 'n foto te skuif.



Rekenaartaal word deur 'n reeks numeriese of alfabetiese kodes ontwikkel en gee aan masjiene bevels hoe om sekere handeling te verrig. Rekenaarkodering is soos 'n gids wat vir die rekenaar presies sê wat en hoe om iets te doen.

'n Programmeerder of kodeerder skryf instruksies wat die inligting en take

beskryf wat nodig is om 'n beeld te maak of 'n handeling uit te voer. Rekenaars **skandeer** die kodes en voer dan die take uit. As jy 'n app of webwerf gebruik, sien jy nie die duisende kodes wat gebruik is om dit te laat werk nie. Ons sien net foto's, prentjies en woorde, maar 'n rekenaarprogrammeerder of -ontwikkelaar het daardie kodes geskryf vir die foto's en teks wat daarop verskyn. ²



Die taal wat rekenaars praat

Programmeringstaal (kodes) help mense om beter met masjiene te kommunikeer.

Jy het seker al van **sagteware** gehoor? Dit is al die programme, roetines en simboliese taal wat die werking van 'n rekenaar se **apparatuur** beheer. Daar is baie, baie, baie kodes nodig vir programme om te werk en daarom het groot maatskappye ook groot spanne wat kodes skryf.



Rekenaarprogramme bestaan uit **datastrukture** en **algoritmes**. Datastrukture bevat die inligting wat die rekenaarprogram moet verwerk. Algoritmes is weer die volgorde van stappe wat 'n program moet volg om daardie inligting te verwerk.

Kom ons kyk na 'n voorbeeld: Maatskappye het gewoonlik salarisprogramme vir personeel. Die program se datastrukture sal personeelinligting, hul werksure en salarisbedrag bevat. Die program se algoritmes het weer die instruksies hoe om elke werknemer se salaris uit te werk en hoe om die salarisstrokies uit te druk.



Kodes word gewoonlik in twee tipe rekenaartale geskryf. Dit word laevlak- en hoëvlaktaal genoem. Laevlaktaal is masjientaal. Dit word ook **binêre** kode genoem en bestaan uit 'n reeks nulle en ene wat gebruik word om instruksies te gee. Elke syfer in 'n reeks kodes word met 'n handeling op jou rekenaar verbind. Hoëvlakprogrammeringstaal herinner aan die taal wat ons praat en die taal word dan na 'n binêre kode verander wat rekenaars verstaan.



Die geskiedenis van kodering

Charles Babbage het in 1822 die meganiese rekenaar (ook verskil-enjin genoem) bekendgestel en dit was die eerste “rekenaar” wat instruksies gekry het. Ada Lovelace was ’n wiskundige en word dikwels die eerste rekenaarprogrammeerder genoem omdat sy beseft het dat syfers meer as net numeriese waarde besit. Sy het ’n algoritme vir die meganiese rekenaar geskryf.

John Von Neumann het in 1945 by die Institute for Advanced Study in Amerika **konsepte** ontwikkel wat die weg vir rekenaartaal gebaan het. Hy het gesê dat **hardeware** eenvoudig moes wees en dan moes daar instruksies vir die hardeware gegee word.



Die eerste hoëvlakprogrammeringstaal vir die verwerking van inligting is flowmatic genoem. Dit is in die vroeë jare vyftig deur Grace Hopper geskryf. Sy was ’n rekenaarprogrammeerder in die Amerikaanse vloot. Rekenaars het toe begin om al hoe belangriker te word.

’n Span wat deur John Backus van die International Business Machines (IBM) Corporation het begin om ’n taal te ontwikkel wat die programmering van wiskundige formules makliker te maak. Die taal Fortran is in 1957 voltooi en het die eerste omvattende hoëvlakprogrammeringstaal geword. Nuwer weergawes daaraan word vandag nog gebruik.



Hoewel die taal goed was om wiskundige bewerkings te doen, was dit nie juis geskik vir ondernemings se rekenaarwerk soos hoe om data te verwerk nie. Verskeie rekenaarvervaardigers, met die ondersteuning van die Amerikaanse regering, het COBOL (Common Business-Oriented Language) in die vroeë jare sestig ontwikkel. Dit het die belangrikste koderingstaal vir gebruik in die ondernemings geword. Nuwe weergawes daarvan word ook nog vandag gebruik.

Proff. John Kemeny en Thomas Kurtz van die Amerikaanse Dartmouth College het ook ’n eenvoudiger weergawe van Fortran ontwikkel. Hulle het BASIC (Beginner’s All-purpose Symbolic Instruction Code), in 1965 ontwikkel en dit was maklik om te leer en maklik om te gebruik. Dit was egter baie stadig en ’n mens kon dit nie

regtig professioneel gebruik nie.

BASIC het wel in die jare sewentig ontwikkel tot 'n koderingstaal wat wêreldwyd gebruik is. In die jare negentig het Microsoft hierdie rekenaartaal aangepas en dit Visual Basic genoem. Dit het 'n gewilde taal vir persoonlike rekenaars geword.



'n Professor in Switzerland Niklaus Wirth het in 1968 die bekende rekenaartaal Pascal ontwikkel. Dit het vandag nog 'n invloed op rekenaartale.

Die LISP-rekenaartaal is in die jare sestig onder leiding van prof. John McCarthy by die Massachusetts Institute of Technology (MIT) ontwikkel. LISP is meestal vir die programmering van **kunsmatige intelligensie** gebruik. Programme wat kunsmatige intelligensie gebruik, probeer om rekenaars meer bruikbaar te maak deur die beginsels van menslike intelligensie in hul programmering te gebruik.

Nog bekende rekenaartale is C wat in die jare sewentig geskryf is en Ada, wat in 1979 ontwikkel is.



Rekenaartale soos Fortran, Ada en C word prosedurele tale genoem omdat dit in verskillende prosedures opgebreek word om verskillende take uit te voer. In die jare negentig het nuwe maniere beskikbaar geraak soos 'n voorwerp-georiënteerde programmering. Die eerste gewilde taal vir hierdie programmering was C++, en is deur Bjarne Stroustrup van Bell Laboratories ontwerp. James Gosling van Sun Microsystems Corporation het in die jare negentig 'n eenvoudiger weergawe van die "C++"-taal ontwikkel. Dis Java genoem is vandag steeds baie gewild.

Vandag bestaan daar honderde tale waarin daar gekodeer word. ³



Kodering se voordele vir kinders

Toe koderingskursusse begin is, is dit meestal bygewoon deur mense wat reeds in die tegnologiewêreld was. Daar is egter gou besef dat kodering baie voordele kan

inhou en vandag is daar selfs koderingsklasse vir kleuterskoolkinders. Dit help kinders om probleme op te los en om soos 'n rekenaar te kan dink. Dit beteken dat jy groot take in kleiner dele kan opbreek om take makliker te voltooi. Dit leer 'n mens ook hoe om foute vinniger op te spoor en na oplossings te soek. Dit kan jou ook leer om meer logies te dink en te volhard. ⁴



Kodering as loopbaan

Daar is verskeie beroepe waarin kennis van kodering nodig is, soos sagteware-ontwikkelaars, webontwikkelaars, kuberveiligheidspesialiste, UX/UI-ontwerpers, masjienleeringenieurs, KI-wetenskaplikes, data-ontleders, robotika-ingenieurs, app-ontwikkelaars, speletjiesontwikkelaars, rekenaarnetwerkargitekte en rekenaarstelselontleders. ⁵



Woordbank

algoritmes	'n Stel reëls vir die berekening of oplossing van probleme, veral ten opsigte van die rekenaar.
apparatuur	Toerusting.
binêre	Tweedelig, tweevoudig.
data	Dit is inligting in 'n vorm waarin dit bewaar en gebruik kan word, veral in 'n rekenaar.
hardeware	Die onderdele en bybehore van 'n rekenaar.
instruksies	Dit is inligting wat vir 'n mens vertel hoe om iets te doen of iets te gebruik.
kodeer	Om volgens 'n kode te skryf.
konsep	Idee, voorlopige plan.

kunsmatige intelligensie	Die afdeling van die rekenaarwetenskap wat daarop fokus om masjiene te laat funksioneer asof hulle oor menslike intelligensie beskik.
rekenaars	Dit is 'n elektroniese apparaat vir die vinnige verwerking en berging van data.
rekenaartaal	Die kunsmatige taal wat ontwerp is om instruksies aan 'n rekenaar te gee.
sagteware	Dit is al die programme, roetines en simboliese taal wat die werking van 'n rekenaar se apparatuur beheer.
skandeer	Aftas.

Artikels oor kodering

- [Wat is kodering?](#)
- [Kyk na die geskiedenis van rekenaartaal](#)
- [Hoekom is kodering belangrik?](#)

Kyk na hierdie video's om meer oor kodering te leer

Hoekom is kodering belangrik?

Wat maak van 'n rekenaar 'n rekenaar?

Hier is 'n 12-jarige ontwikkelaar van *toeps*