

Børn kan også mødes om at kode. Coding pirates er et af fritidstilbuddene (codingpirates.dk).



Start en sjov og kreativ hobby

FÅ BØRNENE TIL AT LEGE MED KODNING

Det er sjovt og kreativt at kode. Mange børn elsker det, når de først får det lært. Her får du basale tips til, hvordan du sætter børnene i din familie i gang med skæg programmering

Vi begynder med en historisk gennemgang. De første højniveausprog – der kom frem i 1950'erne og omfattede for eksempel Fortran, Algol og Cobol – var udviklet som jordnære redskaber uden megen tanke på uddannelse. Det ændrede sig i 1964 med lanceringen af Basic. Bag det sidste navn gemmer sig en forkortelse, Beginner's All-Purpose Symbolic Instruction Code, hvilket understreger sprogets pædagogiske ambitioner. Basic var beregnet til at blive brugt af studerende med et beskedent kendskab til computere. Det overlevede andre sprog fra denne periode og blev brugt i hjemmecomputere i 1970'erne og 1980'erne, og det hjalp endnu en generation med at lære at kode.

Basic var stort set uden konkurrenter i flere årtier. Men hvis man i dag vil hjælpe sine børn med at lære at kode, er det ikke nær så enkelt at vælge et sprog. Der findes i dag adskillige andre sprog, som er udviklet med tanke på uddannelse. Der findes også sprog til begyndere, der bliver betragtet som egnede til uddannelse, selvom det ikke oprindeligt var hensigten med dem.

I denne artikel vil vi forsøge at hjælpe dig med at navigere gennem junglen af programmeringssprog, så du kan beslutte dig for, hvad der er det bedste for dine børns uddannelse. Vi har talt med Carrie Anne Philbin, der er chef for uddannelsessupport i Raspberry Pi Foundation, og som også arbejder med undervisningspen-

summet i det britiske computeruddannelses-program.

Hun har givet os nogle nyttige oplysninger om de forskellige sprog, vi beskæftiger os med her. Hendes indledningsreplik er værd at bide mærke i:

“Jeg tror, at den vigtigste oplysning, som jeg kan give jer, er, at i uddannelse – både formel og uformel – begynder vi ikke med et programmeringssprog. Vi begynder altid med begreber, gerne ofline, før vi bruger redskaber, som hjælper eleverne med at gøre disse begreber mere konkrete i deres bevidsthed.”

SPROG 1: SCRATCH

Hvis man ikke tidligere har beskæftiget sig med uddannelsessprog, er det tænkeligt, at man ikke er stødt på blokbaserede sprog. Udtrykket adskiller sig markant fra “blokstruktureret”, altså det begreb, som er centralt for struktureret programmering og indbygget i de fleste af nutidens programmeringssprog. I modsætning hertil trækker man i et blokbaseret sprog blokke – som man kan betragte som udsagn – fra en palet over på scriptingområdet, hvorefter man redigerer ved at flytte rundt på blokkene i dette område. Det betyder med andre ord, at der kræves meget lidt egentlig skrivning. Det er stort set begrænset til at redigere ting såsom tiden i en venteblok eller afstanden i en bevægelsesblok. Brugervenligheden skyldes ikke mindst, at blokkene omtrent ligner brikker i et puslespil, og det princip mindsker risikoen for, at man kommer til at bruge en blok i en meningsløs sammenhæng.

Scratch er vores første blokbaserede sprog – det er gratis, multiplatform og open source, og det blev udviklet af MIT’s Media Lab – MIT er nærmest svaret på Danmarks DTU. Scratch er det blokbaserede sprog, som Carrie Anne Philbin primært anbefaler. Efter hendes mening er det egnet for børn fra 11 til 13 år før overgangen til tekstbaserede sprog. Til børn fra fem til syv år anbefaler hun det nært beslægtede Scratch Jr. Scratch bliver brugt i formel undervisning og i uformel undervisning i hjemmene.



Scratchs blokbaserede tilgang er måske fremmed for erfarne programmører, men det bliver varmt anbefalet som det første sprog.

Man kan installere Scratch lokalt eller køre det online på <https://scratch.mit.edu>. Scratch Jr. er dog kun tilgængelig som app. Det er beregnet til at blive brugt på tablets, og man kan få det til iPads og Android-enheder.

SPROG 2: BLOCKLY

Blockly er udviklet og bliver hostet af Google. Det er vores andet blokbaserede sprog, og selvom man sommetider møder den antagelse, at det udelukkende sigter på undervisningsøjemed, har dets udviklere ikke kun lavet det til dette formål. Derfor er det også mindre basalt end Scratch og mere egnet til at lave rigtige applikationer. Google skriver specifikt, at “det ikke er noget legetøj,” og at man kan bruge det til at “implementere komplekse programmeringsopgaver.” Vi tillader os at gå ud fra, at det ikke betyder, at Google betragter undervisningsressourcer som legetøj. Til trods for at Blockly er udviklet af Google, er der tale om et open source-projekt.

Ifølge Carrie Anne dominerer Scratch den blokbaserede sektor, og derfor bliver Blockly



BEE-BOT OG BLUE-BOT – KOD SELV EN ROBOT

Til ganske små børn, for hvem brugen af selv et blokbaseret sprog kan være at gå for langt, kan Bee-Bot (650 kroner) eller Blue-Bot (750 kroner) være interessante. De bliver kaldt programmerbare gulvrobotter, og man kan typisk bruge dem med måtter, som robotterne kan programmeres til at bevæge sig på – men man behøver slet ikke at programmere disse enheder med noget sprog. I stedet kan man programmere dem ved at trykke på knapper på robotens krop. Hvis man for eksempel trykker på

“Clear”-knappen efterfulgt af “Forward” to gange, “Right turn”, “Forward” og “Go”, bevæger robotten sig 300 mm (2 x 150 mm) fremad, drejer 90 grader til højre og afslutter med at bevæge sig 150 mm fremad. Der er en kort pause mellem hvert trin. Dette ligner ikke umiddelbart programmering i den normale brug af ordet, men det er faktisk lige netop, hvad det er, og det lærer børnene nogle centrale principper.

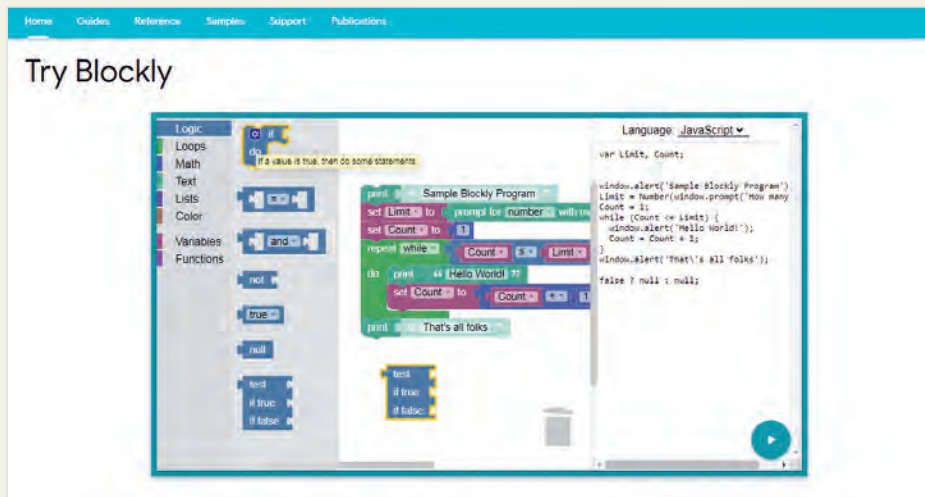
Lad os vende os mod Blue-Bot, som man kan bruge

på samme måde som med Bee-Bot. For det første er den gennemsigtig, således at nysgerrige sjæle kan få en fornemmelse af den hardware, som styrer den.

For det andet har den et Bluetooth-interface. Børnene kan styre robotten på afstand med en håndholdt enhed, dog kun via Windows, iPad og Android. Kort sagt gør den gratis app det muligt at lave et program på skærmen med en enkel proces, downloade dette program til Blue-Bot og eksekvere det.



Med Bee-Bot kan børn på en sjov måde programmere en robot ved at trykke på de forskellige knapper på dens krop.



Googles Blockly er blokbasert, men det kan bruges til at oprette rigtige applikationer. Det er bedre egnet som undervisningsprog til ældre børn, end Scratch er.

anvendt, når det er nødvendigt, for eksempel ved programmering i App Inventor (<https://appinventor.mit.edu>) eller med enkeltkort-computeren Micro:bit, hvor Scratch oprindeligt blev understøttet. Blockly ligger i sin struktur tættere på et tekstbaseret sprog. Af den årsag er det egnet til børn, der er mellem 10 og 14 år, og det bliver ikke anbefalet til yngre aldersgrupper.

Blockly kører i en browser fra <https://developers.google.com/blockly>, og det omfatter en interessant mulighed for at generere og vise koden i en række tekstbaserede sprog – nemlig JavaScript, Python, PHP, Lua og Dart – der er i overensstemmelse med den blokbaserede kode.

Vi gætter på, at det kan have en vis værdi, når man skal migrere fra et blokbasert sprog til et tekstbaseret sprog. Python og JavaScript er oplagte valg i denne sammenhæng, men Lua og Dart kan virke lidt overraskende. Selvom Scratch og Blockly ikke er de eneste blokbaserede sprog, er det dem, vi hører langt mest om, når det gælder undervisning. Nu vender vi vores opmærksomhed mod de tekstbaserede

sprog, som er dem, de fleste af os nok kender mere til.

SPROG 3: BASIC-256

Når vi taler om et sprog, der er 56 år gammelt, som ikke har fulgt mange af de nyere programmerings-paradigmer, og som bliver udsat for nogen kritik, kan vi tvivle på, om vi bør inddrage Basic. Imidlertid bliver det stadig brugt kommercielt, om end i form af Visual Basic, og det bliver stadig anbefalet som et sprog til børn. Derfor er det passende at tage det med her. Basic-256 er måske den hyppigst citerede moderne (i hvert fald halvmoderne) implementering til undervisningsbrug, og det er derfor den version, vi vil se på. I modsætning til de oprindelige Basic-kommandolinje-implementeringer er denne version mere et integreret udviklingsmiljø (IDE) med vinduer til koden, tekst-output og grafisk output.

Denne udgave af Basic blev udviklet i 2007 specifikt til undervisning i programmering. Ligesom det oprindelige Basic, men ulig nogle af de nyere versioner, er styringen af forløb stort set begrænset til GOTO, GOSUB, FOR...NEXT og IF...THEN...ELSE...ENDIF. Det bidrager til sprogets enkelhed, men det bliver også mødt med nogen kritik. Nogle mener, at det ikke indlærer principperne bag struktureret programmering, men i stedet fremmer "spaghettikode". Det bliver modsagt af mange mennesker, der siger, at de har lært at programmere i Basic, og at det ikke forhindrede dem i at få en vellykket karriere inden for softwarekodning med mange forskellige sprog.

SPROG 4: LOGO

Et andet undervisningsprog, som ikke ligefrem er nyt, er Logo. Der findes adskillige versioner, som kører lokalt, og UCLogo nyder almindelig respekt. Man kan få adgang til sproget online, for eksempel på www.transum.org/software/Logo og www.calormen.com/jslogo. Logo er et komplet programmeringssprog, og det bliver beskrevet som "multi-paradigme", selvom dets anvendelse i undervisning har haft en tendens til at koncentrere sig om kun et element af sproget. Det drejer sig om "skildpadde"-funktionen, der styrer en skildpaddes bevægelser og tegner dens færden på skærmen og dermed producerer et grafisk output.

Der er kun nogle få skildpadde-statements, men Logo tilbyder en enkel introduktion til tekstbaseret programmering.

Det kan måske være et første skridt videre end den blokbaserede tilgang – muligvis for børn, der er langt yngre end de 11 år, som er den anbefalede begyndelse på tekstbaserede sprog.

Hertil kommer, at programmering med henblik på at lave grafisk output appellerer til de ganske unge.

Vi har imidlertid haft nogen tvivl om grunden til, at sproget får mindre opmærksomhed, end det gjorde for få årtier siden. Kan det skyldes, at moderne sprog egner sig bedre til netop denne niche?

Carrie Anne er bestemt entusiastisk, når det gælder brugen af skildpadder. "Brugen af skild-

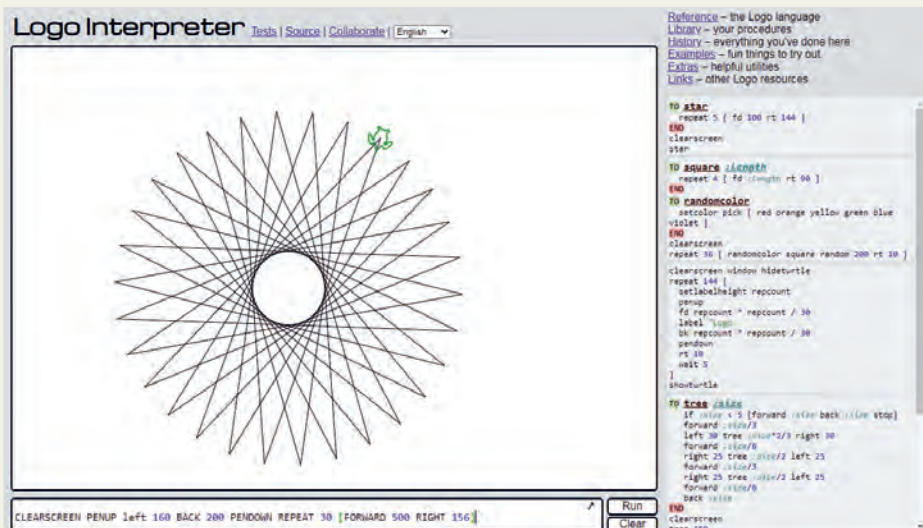
SONIC PI – DIGITAL JAMSESSION

Sonic Pi (<https://sonic-pi.net>) bliver beskrevet som et kodebaseret performance-redskab. Det kører under Raspberry Pi OS (det er udviklet i samarbejde med Raspberry Pi Foundation) og med Windows og macOS. Det bliver også kaldt et live-kodningsmiljø. Det vil sige, at performerne kan skrive kode løbende og lave musik, mens de optræder. Man kan tale om at etablere en digital jamsession. Der er imidlertid forskelle, idet det er almindeligt at vise livekoden

på en skærm under forestillingen, således at publikum kan være med til kodningsoplevelsen.

Sonic Pi bliver brugt af professionelle kunstnere og navnlig til børn. Applikationens udvikler, Sam Aaron, der er skribent på magasinet MagPi, siger om værket: "Sonic Pi blev udtænkt og gennemført i klasseværelser i tæt samarbejde med lærerne, og projektet har nydt godt af mange design-iterationer på baggrund af hundreder

af timers elevobservationer. Enkelhed er konstant blevet bygget ind i den centrale designfilosofi, som består i udelukkende at inddrage funktioner, der er nemme at forstå for et barn på ti år, og som det er nemt for læreren at bringe dette barn." Så vidt projektets enkelhed, men det er vigtigt at erindre, at det bygger på programmeringssproget Ruby, og det vil sige, at der ikke er nogen grænser for mulighederne.



Logo er langtfra født i går, men det har stadig sine fans, og hvis man kun ser på "skildpadder"-instruktionerne, kan det være ideelt som det første tekstbaserede sprog til børn.

paddinger er meget udbredt ved undervisning i programmering. Fra at bruge Bee-Bots i folkeskolen med henblik på at forstå sekvenser og algoritmer til at bygge Raspberry Pi-robotkøretøjer for at komme igennem mere komplekse labyrinter."

SPROG 5: PYTHON

Med vores næste sprog går vi meget længere end de sprog, der først og fremmest blev lavet til undervisning. Dette sprog har generel relevans i mange sammenhænge. Python er i dag det tredjemest brugte sprog, og det kunne være nok til at sikre det en plads i undervisningen, men der er mere end som så. Python bliver i almindelighed anbefalet til uddannelsesformål – og i særdeleshed som det første tekstbaserede sprog. Det skyldes navnlig, at det er meget enklere at lære end mange andre sprog.

Carrie Anne mener, at dets brugervenlighed ikke er den eneste faktor bag sprogets udbredelse inden for undervisning. Blandt de andre grunde til Pythons popularitet er, at det er gratis, tværsplatform og open source. Hun

fremhæver også de mange IDE'er, der er blevet udviklet i årenes løb, og som lever op til undervisningens krav, for eksempel Mu (<https://codewith.mu>), der bliver beskrevet som en simpel Python-editor til begynderprogrammører, og Trinket (<https://trinket.io>), der er browser-baseret).

Ressourcer til lærere er også vigtige, hvilket forklarer brugen af programmet i skoler og underetager, at Python er egnet til hjemmearbejde.

"Hos NCCCE (det britiske computeruddannelsesprogram, red.) bygger meget af vores professionelle udvikling på Python. Det betyder, at der er stor sandsynlighed for, at man også vil bruge det som det tekstbaserede sprog i klasseværelset."

Hun er også inde på, at andre ressourcer til at hjælpe lærere i overvældende grad bliver lavet med udgangspunkt i dette sprog, og disse ressourcer er ofte frit tilgængelige, for eksempel Raspberry Pi-projekter (<https://projects.raspberrypi.org/en>). Python anbefales til børn på 11 år og opefter.

SPROG 6: C++

C++ er stadig vidt udbredt. Det er faktisk kun et skridt efter Python i det velansete TIOBE-indeks (en organisation, der vurderer kvaliteten af softwarekodning) og dermed på en fjerdeplads. Det er en af grundene til, at det bliver anbefalet som et egnet sprog til undervisningsbrug. Nogle hævder, at det er så vidt udbredt, at kendskab til C++ giver eleverne gode jobmuligheder eller et udmærket grundlag for videre uddannelse.

Andre undervisere mener, at det er et ekstra plus, at det giver eleverne en god forståelse for, hvordan computere

fungerer. Grunden er, at det i langt højere grad end de fleste andre sprog, der bliver anbefalet til undervisning, er et lavniveau-sprog. Det er nemlig kun et enkelt skridt op i forhold til den grundlæggende assembly-kode.

ANDRE SPROG

Selvom vi har dækket de programmeringssprog, som er de mest udbredte i undervisning, er de ikke de eneste. Det er værd kort at nævne nogle af alternativerne. Faktisk kom Carrie Anne ind på et par stykker.

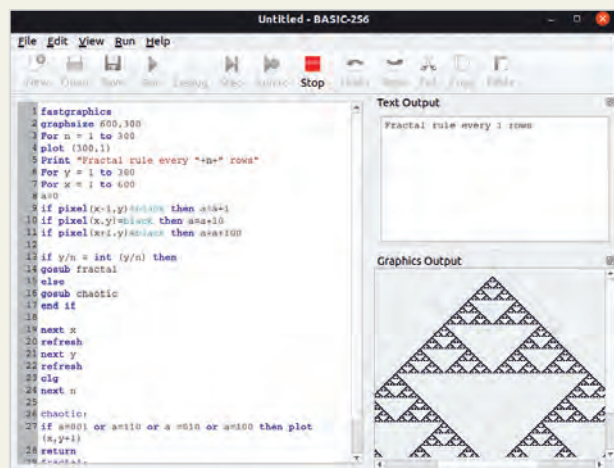
Hendes første forslag er formentlig ret overraskende, eftersom sproget kun er på en tiendeplads i TIOBE-indekset, som registrerer programmeringssprogs popularitet.

Man må imidlertid erkende, at eftersom for eksempel Scratch er nede på nummer 21, er det jo ikke til at komme uden om.

"Ruby bliver stadig mere populært inden for undervisning. Sonic Pi er et godt eksempel på et lettilgængeligt redskab, som kan hjælpe brugen til at kode musik, og som også bliver brugt i forbindelse med indlæring af programmeringsbegreber til unge mennesker – på baggrund af Ruby. Som sprog er det tilgængeligt, og dets enkle syntaks gør det endnu mere tilgængeligt for børn og unge mennesker," siger hun.

Vi føler en stærk trang til at konkludere, at Carrie Anne har fuldstændig ret, når hun siger, at det er vigtigere at forstå begreberne end at lære specifikke sprog. Når det er sagt, må vi også nævne, at naturligvis skal man overveje fordele og ulemper ved de forskellige sprog, vi har nævnt, og man er nødt til at undgå at introducere sine børn for sprog, der ikke er egnede til deres alderstrin.

Det er også nødvendigt at tænke over, hvilke sprog man kender godt nok til at undervise i dem. Men selvom man for nogle har været beskæftiget med at vedligeholde ældre systemer og derfor er velbevandret i forrige tiders sprog, er vi ikke sikre på, at en anbefaling af undervisning i Cobol til ens femårige søn vil være den klogeste vej at tage de første skridt på. **ADD**



Basic har afgjort sine kritikere, men på trods af kontroverser bliver Basic-256-dialekten alligevel anbefalet til undervisningsformål.