

Steen Petersen
Fredericia Realskole
Danmark
Science on Stage 2024



Är Scratch lösningen för ett lekfullt och undersökande sätt att ta sig an modelleringskompetens inom naturvetenskap?

Hur skriver vi en kod i Scratch som visar hur molekyler rör sig i gasform? Eller en modell av ett svart hål? Eller den kinetiska energin hos en pendel?

Det här läromedlet börjar med en introduktion och bakgrunden för hur man programmerar simuleringar i Scratch och varför det är logiskt att elever själva skriver kod när de övar på undersökande kompetenser och modelleringskompetenser inom naturvetenskap. Med ett lekfullt, utforskande och undersökande sätt att ta sig an uppgiften har eleverna möjligheten att skriva kod i Scratch för en eller fler av simuleringarna, baserat på instruktioner i läromedlet. I läromedlet diskuteras de akademiska, praktiska och didaktiska aspekterna av forskningsbaserad undervisning, och mer än tjugo koder för Scratch och BBC Micro:bit. Fånga dina elevers förståelse för modeller och teknologi.

Det är en fördel om du redan är registrerad som användare på scratch.mit.edu och har lite erfarenhet av att programmera i Scratch. Eller om du bara är väldigt nyfiken.

Det här läromedlet består av en kort introduktion och en exempelkod i Scratch. Materialet är riktat till elever i de övre klasserna (12–16 år). Läromedlet innehåller koder för att animera en regnbåge, vattnets kretslopp, ett svart hål, exoplaneters rörelse runt en stjärna, Niels Bohrs atommodell, pendelsvängning, en liten magnetmodell, regelbundna polygoner, Jordens magnetiska fält, kedjereaktioner och mycket mer.

Använd BBC micro:bit som ett mätinstrument vid fysikundervisning. Läromedlet innehåller också koder för BBC micro:bit. Materialet innehåller instruktioner för hur du använder micro:bit som en termometer, för att mäta kinetisk energi hos en svängande pendel och som ett instrument för att mäta magnetfält.

Steen Petersen, som deltog i Science on Stage 2024 tillsammans med den danska gruppen, är lärare i matematik, fysik, kemi, geografi och biologi på Fredericia Realskole. Han är själv väldigt nyfiken på programmering som en del av modelleringskompetenserna inom naturvetenskap.

I Danmark är "modelleringskompetens" en av fyra kompetenser inom undervisningen i naturvetenskap. Genom att programmera i Scratch skapar eleverna sina egna naturvetenskapliga modeller. De får en ökad förståelse för akademiskt material genom att skapa koder för små animeringar och simulera fysikaliska, matematiska och kemiska fenomen på en dator.

Härnäst används termerna programmering, kod, animering, digital modell, simulering och datalogiskt tänkande. Ibland kan termernas betydelse överlappa varandra. Jag använder ordet programmering i betydelsen att skriva ett program i Scratch. Och jag använder Scratch som exempel.

Animering och simulering är ord som syftar på att skapa en digital modell av ett fysikaliskt fenomen på en dator. Det kan till exempel vara en simulering av var elektronerna är placerade i Bohrs atommodell. På dataskärmen visar simuleringen hur elektronerna placerar sig och först fyller upp det inre elektronskalet. Eller en simulering av hur små magneter rör sig när ett stålblad magnetiseras.

Att programmera koder ger nya möjligheter för eleverna att aktivt delta i "sin egen inläring". Frågan "vad händer om?" och att vara tvungen att göra programmeringen långsamt och tänka igenom och justera programmet, såväl som att diskutera möjliga förbättringar i koden. Det tar tid och den långsamma processen är viktig när eleverna jobbar för att få programmet att fungera och förklarar för varandra vad modellen visar.

Eleverna deltar aktivt när de skriver små program själva. De lär sig både om bekanta fysikaliska fenomen och samtidigt hur program i Scratch fungerar. Eleverna jobbar med instruktionsbladet i det här häftet. Instruktionerna hjälper eleverna att komma igång med grundläggande kodning i Scratch. Därefter finns det möjligheter att diskutera och fortsätta utveckla modellerna.

Eleverna ställs inför utmaningar längs vägen när de försöker förutsäga vad programmet gör, testa programmet, och de undersöker om det motsvarar deras förväntningar, de ändrar och anpassar variabler, tolkar resultat och skriver till slut nya program.

Det kan ta lång tid att få ett program att fungera perfekt. Det är just det långsamma i processen som är det viktiga för eleverna, eftersom de verkligen behöver förstå och förklara vad som sker. Det tar tid att skriva koden, det tar tid att avbugga koden, testa och justera. Den här processen är viktig.

När vi löser uppgifter i naturvetenskapliga ämnen genom programmering kan det bli ännu tydligare att vi jobbar med vetenskapliga modeller. Modeller är förenklingar av verkligheten, och eleverna är inte alltid medvetna om att de jobbar med förenklingar med begränsningar och antaganden. Att förstå naturvetenskap som obestridlig fakta kan vara begränsande när det gäller hur eleverna lär sig naturvetenskapliga metoder och sätt att tänka. (fritt översatt från Naturfag, nr. 1, 2022, s. 67, Andreas Haraldsrud, Universitetet i Oslo)

Programmering och simulering inom naturvetenskap är nära besläktade med ett av kompetensmålen inom ämnet teknologiförståelse.

Datalogiskt tänkande behandlar analys, modellering och strukturering av data och processer för data. Det här betyder att eleverna måste lära sig att avkoda fenomen och processer i det dagliga livet, i professionell kontext och i digitala artefakter och beskriva dem i form av algoritmer och digitala modeller. Källa: Teknologiförståelse, emu.dk)

Angående datalogiskt tänkande så anges det i läroplanen för teknologiförståelse under färdighets- och kunskapsmål att eleven måste tillägna sig kunskap om tekniker för att konstruera och utvärdera digitala modeller, och att eleven ska kunna konstruera och utvärdera digitala modeller av fysikaliska, kemiska och teknologiska omgivningar.

Det här häftet består av arbetsblad för tjugo program för modeller i Scratch och BBC micro:bit. Bladen beskriver kort idén med programmet, visar ett förslag till en scratchkod och föreslår under "diskussion" hur eleverna själva kan förbättra, ändra och fortsätta utveckla programmet. Diskussionen skapar också ramarna för en diskussion om vad den digitala modellen är bra och mindre bra på att förklara.

Arbetsbladen hjälper eleverna få kunskap om hur man konstruerar digitala modeller och bedöma och diskutera digitala modeller i den fysikaliska, kemiska och teknologiska världen.

Att undervisa i naturvetenskap genom att koda simuleringar kan gå till enligt följande:

- 1. Dela ut utskrivna arbetsblad till eleverna och börja med att tillsammans förutsäga vad koden gör.**
- 2. Skriv koden i Scratch eller BBC Micro:bit och gör ett test.**
- 3. Undersök om programmet kan förbättras genom att ändra variabler, lägga till nya pusselbitar eller ta bort en pusselbit.**
- 4. Fortsätt utveckla koden eller skriv en ny kod. Längst ner på arbetsbladen finns ett upplägg för diskussion och för att fortsätta undersöka modellen.**

Scratch (scratch.mit.edu) är en plattform där elever kan programmera, rita former, få former att röra sig på skärmen, skapa ljud, rita med en penna, göra dataspel och mycket mer inom kodning. Programmeringen sker genom att eleverna drar in olika pusselbitar och sätter ihop dem i programmeringsfältet.

Scratch är gratis att använda. Läraren kan skapa ett lärarkonto med sina klasser, och där kan eleverna registrera sig som användare. Använd aldrig ditt eget namn som användarnamn.

Jag skrev största delen av koderna själv. Vissa koder för en gasmodell, pendel och temperaturmätning fick jag inspiration till av andra. Jag hänvisar till källor nedan. Det ligger inte i mitt intresse att kränka andras rättigheter.

Steen Petersen, April 2024



Se på Youtube

Molekyler i gasform	https://scratch.mit.edu/projects/690148787
Niels Bohrs atommodell	https://scratch.mit.edu/projects/690475256
Pendelsvängning	https://scratch.mit.edu/projects/690530194
En liten magnetmodell	https://scratch.mit.edu/projects/693361885/
Rita en regnbåge	https://scratch.mit.edu/projects/694989406/
Jordens magnetiska fält	https://scratch.mit.edu/projects/695543196/
Regelbundna polygoner	https://scratch.mit.edu/projects/694973251/
Modell av ett svart hål	https://scratch.mit.edu/projects/699037357/
Vattnets kretslopp	https://scratch.mit.edu/projects/710779461
Solsystemet	https://scratch.mit.edu/projects/717621085/
Exoplanet i omlopps bana	https://scratch.mit.edu/projects/721395361/
Snökristall	https://scratch.mit.edu/projects/614727439
Trianglar	https://scratch.mit.edu/projects/132609162
Spiraler 1	https://scratch.mit.edu/projects/194023685
Spiraler 2	https://scratch.mit.edu/projects/194023685
Mandala	https://scratch.mit.edu/projects/133081830

BBC micro:bit kinetisk energi hos en pendel

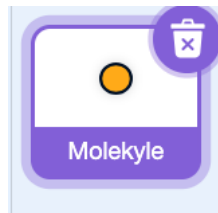
BBC micro:bit bygg en kompass

BBC micro:bit temperatur del 1 och del 2

BBC micro:bit magnetfältsmätare

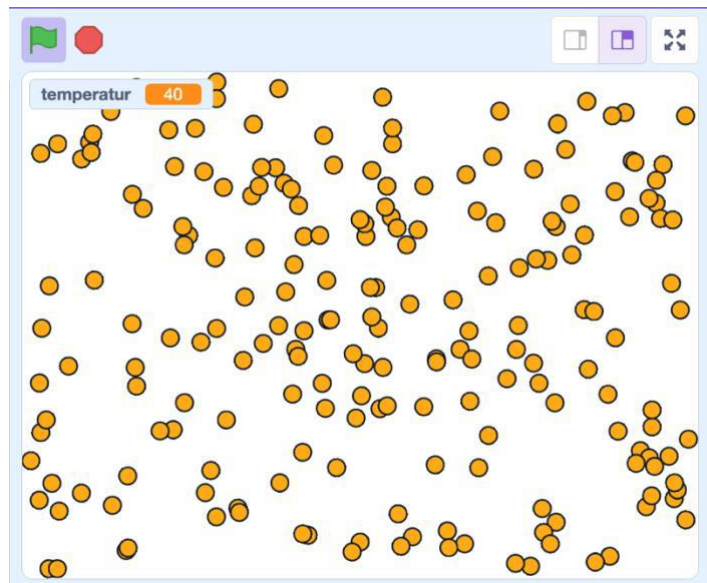
Molekyler i gasform

Rita en ny sprite som en molekyl.



Skriv koden.

Skriv in en variabeln och kalla variabeln "temperatur".



	<p>Programmet frågar efter temperaturen. Skriv ett slumpmässigt tal mellan 0 och till exempel 100.</p> <p>Då skapas 30 kloner som rör sig slumpmässigt på skärmen med en slumpmässig riktning mellan 0 och 360 grader och en hastighet som är kvadratroten ur temperaturen.</p> <p>De studsar tillbaka om de träffar kanten.</p> <p>Programmet fortsätter på obestämd tid.</p>
--	--

Undersök:

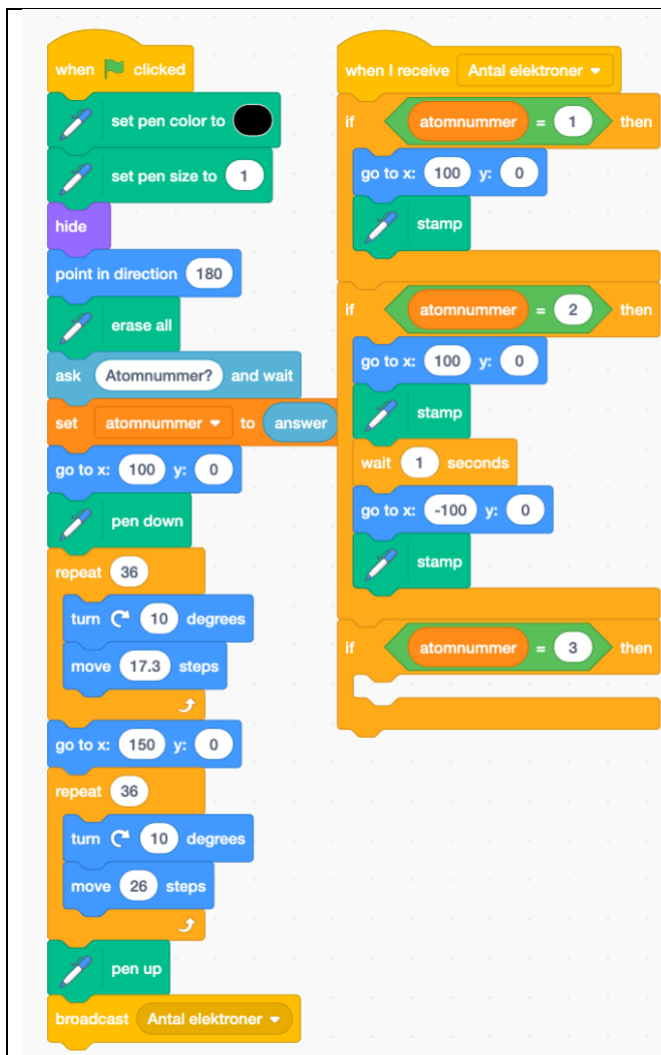
Hur skapar vi en simulering som kan visa de andra aggregationstillstånden: fast form och vätska?

Niels Bohrs atommodell

I Niels Bohrs atommodell har en atom elektroner i skal runt atomkärnan. Elektronerna fyller skalen från insidan, och det innersta skalet kan innehålla två elektroner. Skal nummer 2 har plats för åtta elektroner.

Programmet frågar efter atomnumret. Sedan ritas två cirklar. Om atomnumret är 1 placeras en elektron i det första skalet. Om atomnumret är 2 placeras två elektroner i det första skalet. Fortsätt programmet själv med fler elektroner i det andra skalet.

I det här programmet behöver du en förlängning. Du måste hitta **pennan** som används för att rita.



Elektronerna visas som gröna cirklar i det första och andra skalet.

Du måste fortsätta koda själv så att elektronerna visas när det atomiska numret är 3.

Ställ in riktningen, Direction = 180 (ner)

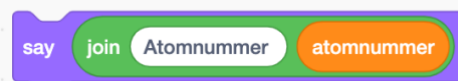
Bakgrunden är ett koordinatsystem som kan hjälpa dig med koordinaterna.

Koordinat för elektron nummer 3 (150, 0)

Koordinat för elektron nummer 4 (0, 150)

Undersök:

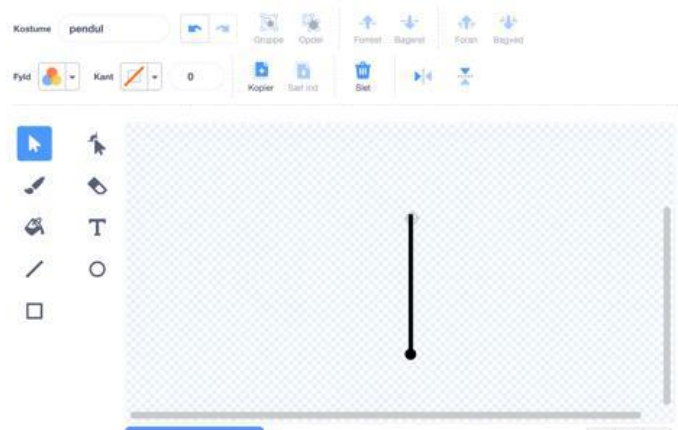
Hur får du programmet att säga atomnumret?



Pendelsvängning

Rita pendeln i en ny sprite.

För att sätta igång pendeln ska du använda musen för att lyfta pendeln på skärmen ut mot ena sidan.



The image shows two Scratch scripts for a pendulum simulation. The first script, on the left, is triggered by a 'when clicked' event. It sets the pendulum speed to 0 and the mouse hold variable to 0. It then enters a 'forever' loop. Inside the loop, there is an 'if' statement: if 'mus hold' is 1, the sprite points towards the mouse pointer and turns 90 degrees; otherwise, the pendulum speed is changed by the cosine of the direction multiplied by 5, the speed is then multiplied by 0.95, and the sprite turns by the pendulum speed degrees. The second script, on the right, is also triggered by a 'when clicked' event and enters a 'forever' loop. It checks if the mouse pointer is touching the stage and the mouse is down. If true, it sets 'mus hold' to 1, waits until the mouse is no longer down, and then sets 'mus hold' back to 0.

Pendeln demonstrerar en dämpad svängning där den dämpas rätt snabbt, vilket får pendeln att svänga mindre och mindre tills den slutligen stannar.

Undersök:

Vad händer om du ändrar faktorn 0,95 till ett annat tal?



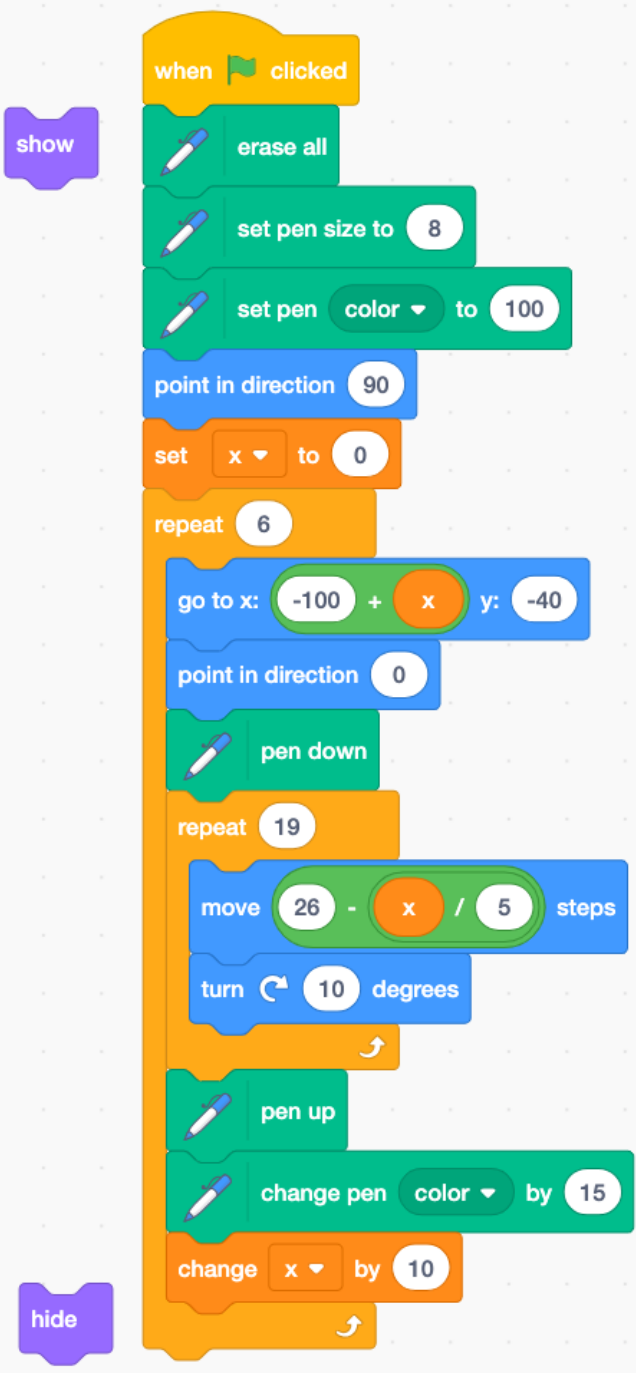
Förklara formeln.

$$T = 2 * \pi * \sqrt{\frac{l}{g}}$$

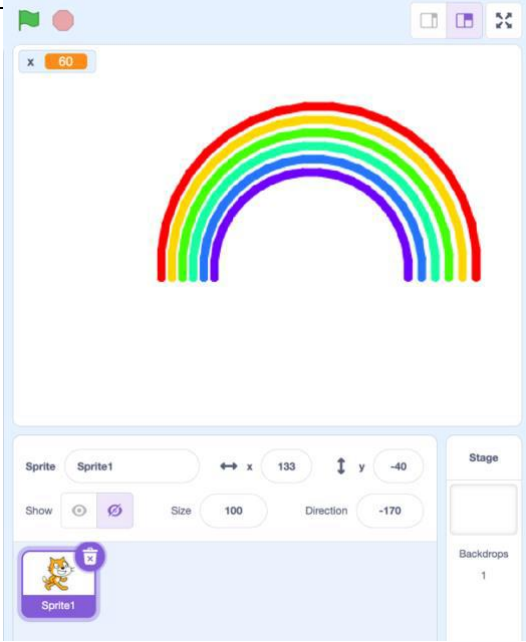
Rita en regnbåge

Färgerna i en regnbåge skapas när ljuset från solen bryts i vattendroppar. Regnbågens färger skapas eftersom de olika färgerna i solljuset bryts i lite olika vinklar.

Skriv ett program som ritar färgerna i en regnbåge.



```
when green flag clicked
  show
  erase all
  set pen size to 8
  set pen color to 100
  point in direction 90
  set x to 0
  repeat 6
    go to x: -100 + x y: -40
    point in direction 0
    pen down
    repeat 19
      move 26 - x / 5 steps
      turn 10 degrees
    pen up
    change pen color by 15
    change x by 10
  hide
```



Undersök:
Vad händer om du ändrar vissa av talen och färgerna?

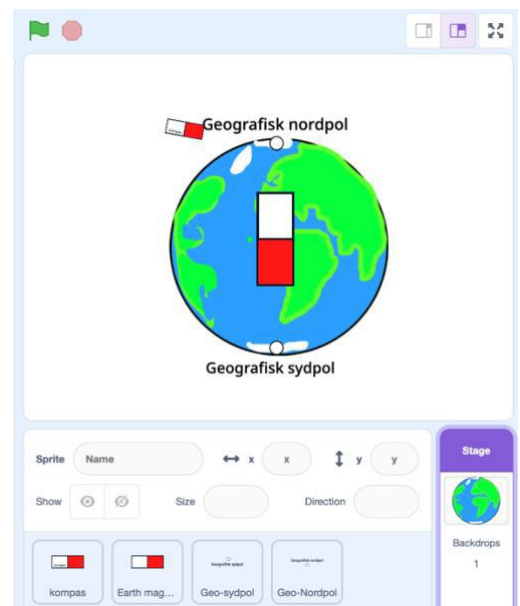
Jordens magnetiska fält

I den här modellen av Jordens magnetiska fält föreställer vi oss att det finns en stor stavmagnet inuti Jorden.

Stavmagneten inuti Jorden pekar med den magnetiska sydpolen upp mot Jordens geografiska nordpol.

Då pekar kompassen med sin nordpol mot Jordens geografiska nordpol.

Jorden ritas som bakgrund. Två små cirkclar används som den geografiska nord- och sydpolen.



	<p>Kompassen är den lilla stavmagneten utanför Jorden och den pekar alltid mot Jordens inre magnet.</p> <p>Kompassen kontrolleras med piltangenterna.</p> <p>Om kompassen är ovanför ekvatorn (och värdet på y är större än 0) måste kompassen peka mot den geografiska nordpolen.</p> <p>Om kompassen är nedanför ekvatorn (och värdet på y är mindre än 0) måste kompassen peka mot den geografiska sydpolen.</p>
	<p>Den stora magneten är en modell av Jordens magnetfält.</p> <p>Med mellanslagstangenten kan du visa och gömma magneten. Kostym1 är en bild av magneten. Kostym2 är tom.</p> <p>Den har två kostymer där den första kostymen är en magnet och den andra är tom.</p> <p>”Point in direction 180” betyder att magneten dras horisontellt i ritkoden, men måste stå vertikalt.</p>

Undersök:

Varför pekar sydpolen på stavmagneten inuti Jorden mot den geografiska nordpolen?

Hur fungerar en kompass?

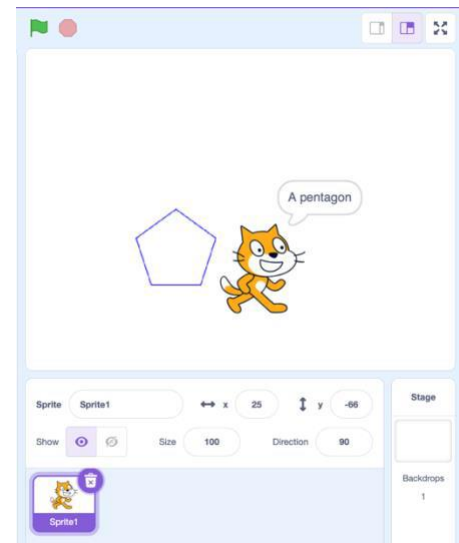
Vad är modellen bra eller dålig på att förklara?

Regelbundna polygoner

En regelbunden polygon är en form med sidor som är lika långa och vinklar som har samma mått.

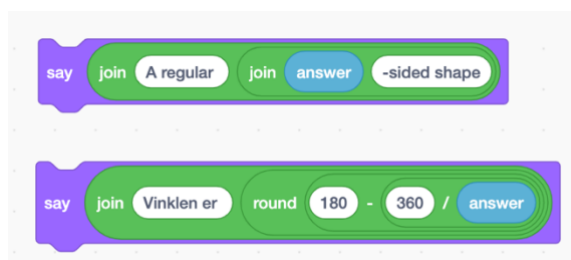
En regelbunden femhörning har fem sidor som är lika långa och fem vinklar med samma mått.

Här är ett program som konstruerar en regelbunden polygon med ett antal sidor som du skriver in som variabeln 'answer'.



<p>Sidans längd justeras i beräkningen: $\text{sidlängd} = 60 - \text{answer}$ för att se till att en stor polygon ryms på skärmen.</p> <p>Två gröna konnektorer läggs ihop i den sista pusselbiten i programmet för att sätta ihop orden 'Här är en [answer]-sidig form'.</p> <p>Nedan ser du hur du kan använda två konnektorer för att få texten på skärmen.</p>	A Scratch script starting with "when green flag clicked". It includes: "erase all", "go to x: -100 y: -85", "point in direction 90", "erase all", "ask 'Number of sides in the polygon?' and wait", "pen down", a "repeat" loop with "answer" iterations containing "move 60 - answer steps" and "turn 360 / answer degrees", "pen up", "go to x: 25 y: -66", and a "say" block with "join A regular", "join answer", and "-sided shape".
--	---

Undersök:

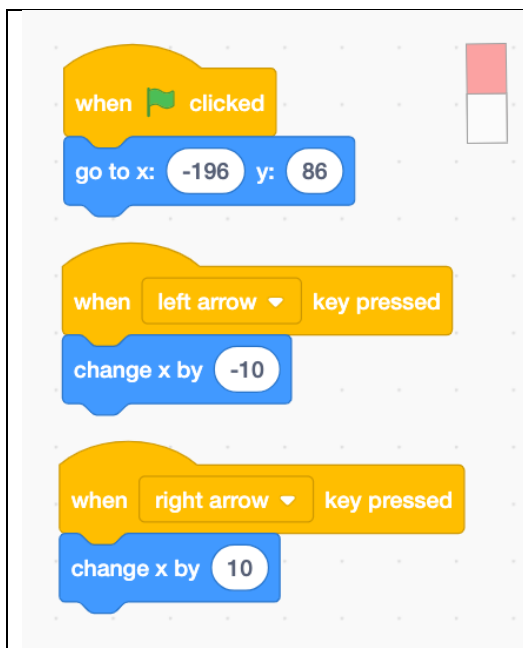
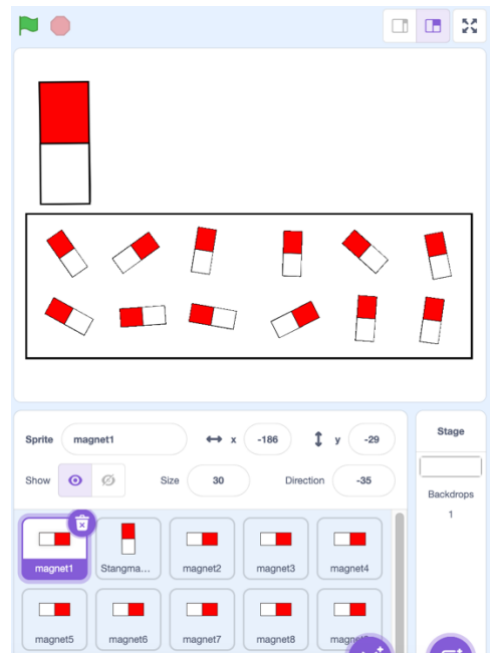


En liten magnetmodell

Magnetiseringen av ett stålblad förklaras genom den lilla magnetmodellen.

Många små magneter inuti stålbladet följer en stavmagnet som långsamt flyttas längs stålbladet. När de små magneterna följer stavmagneten magnetiseras stålbladet.

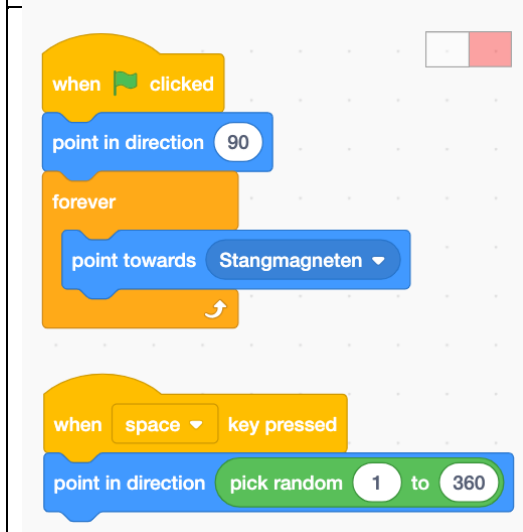
Flytta stavmagneten genom att använda den högra och vänstra piltangenten.



Rita en stavmagnet i en ny sprite.

Skriv koden.

Använd den vänstra och högra piltangenten för att flytta stavmagneten.



Rita en liten magnet i en ny sprite, skriv programmet och gör sedan så många kopior som du behöver.

Undersök:

Vilken del är modellen bra på att förklara?

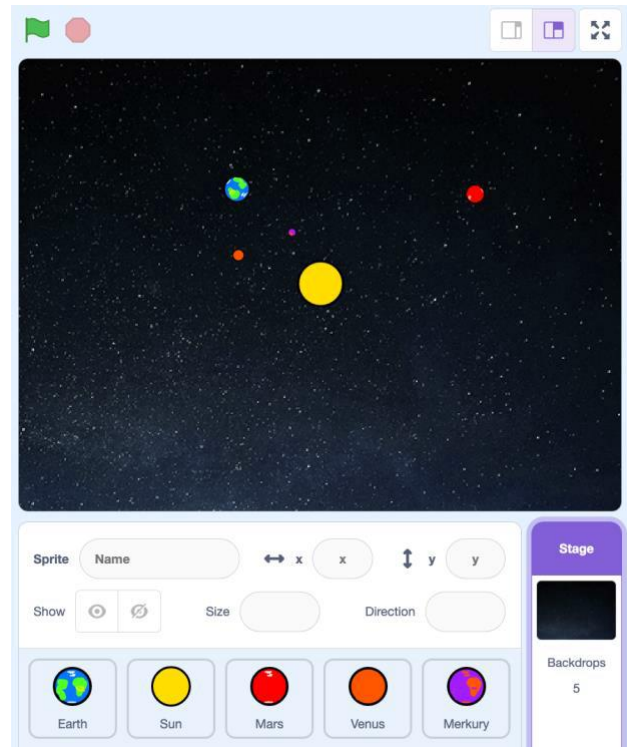
Vilka delar av magnetiseringen kan modellen inte förklara?

Solsystemet

Skriv en kod till en simulering av solsystemet. Simuleringen visar de fyra inre planeternas omloppsbana runt solen i solsystemet.

Använd stjärnor som bakgrund.

Skapa fem nya sprite och rita de inre planeterna.



Omloppsbana

Merkurius

88 dagar

Venus 225 dagar

Jorden 365 dagar

Mars 687 dagar



Koden för Jordens omloppsbana runt solen.

Använd samma kod för Merkurius, Venus och Mars genom att ändra variablerna för startpositionen och omloppsbanan.

Undersök:

Vad är modellen bra/dålig på att visa?

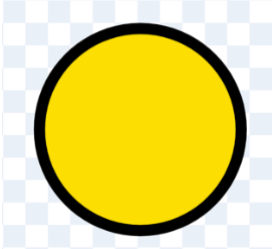



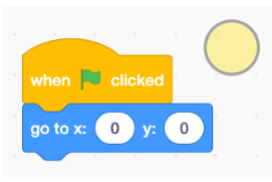

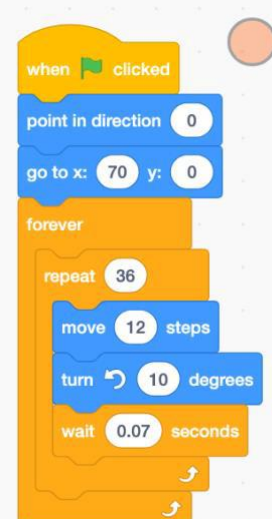
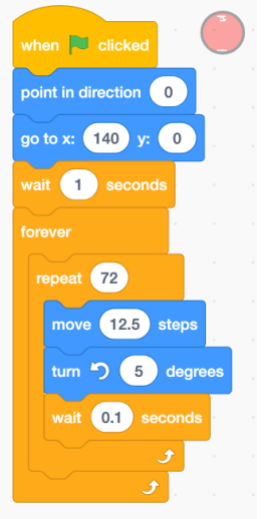
Undersök avstånds- och aspektförhållandet mellan solen och planeterna.

Solsystemet 2

Här är några tips för koderna för planeterna Merkurius, Venus och Mars.

Skapa nya sprite och rita dina egna planeter i Costumes.



Solen	Merkurius	Venus	Mars
			
<p>Solen placeras i mitten av skärmen.</p> 			

Omloppsbanan
Merkurius 88
dagar
Venus 225 dagar
Jorden 365 dagar
Mars 687 dagar

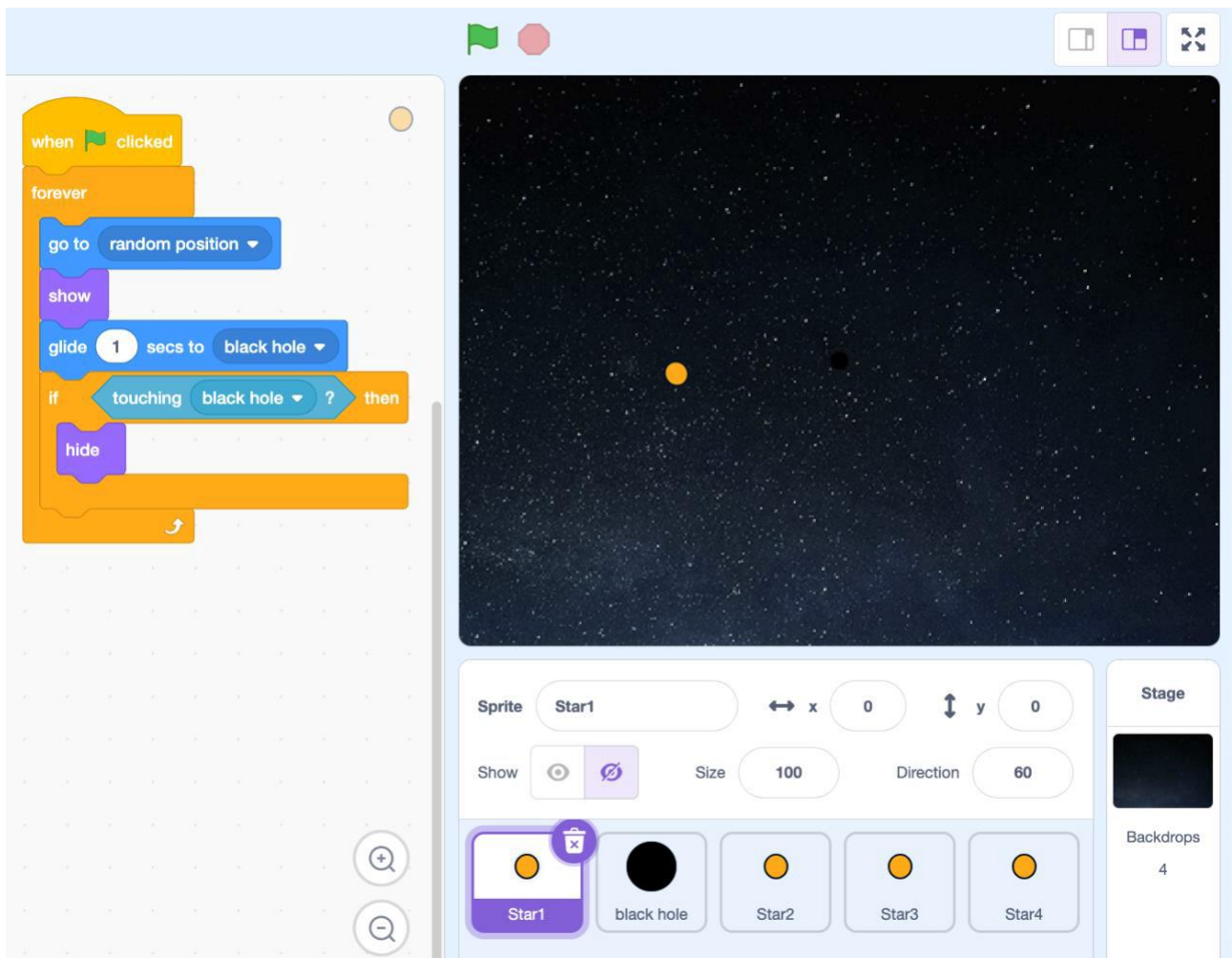
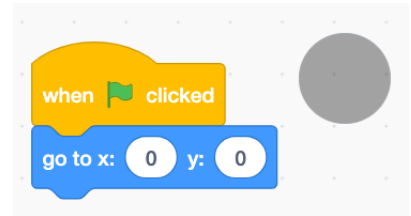
Modell av ett svart hål

Universum består av planeter, stjärnor, galaxer och svarta hål. Svarta hål är en av de märkligaste sakerna i universum som många astronomer förundrar sig över.

De är så tunga att de attraherar all materia. Inte ens ljus kan undslippa ett svart hål.

I modellen bygger du ett svart hål som sväljer gula stjärnor. Gör fyra kopior av star1 och låt dem börja lite senare.

Det svarta hålet i koden är en liten svart punkt i mitten.



Undersök:

Astronomer har fotograferat svarta hål. Sök på nätet och ta reda på mer om svarta hål. Gas och stjärnor nära kanten av ett svart hål lysas upp och "sväljs" av det svarta hålet.

Vad väger ett svart hål?

Var kommer svarta hål ifrån och hur bildas de?

Vad är en händelsehorisont?

Vattnets kretslopp

Det finns många teckningar och illustrationer av vattnets kretslopp i fysikböcker, geografiböcker och på nätet.

Du kan använda bilderna som inspiration till landskapet och skapa dina egna symboler för regn, avdunstning och cirkulation.

Nu ska du göra en liten bildserie som visar vattnets kretslopp.

Du kan förklara några av de här termerna i bildserien: Avdunstning, kondensering, nederbörd, infiltration, grundvatten, ytvatten, regn, snö och solenergi.

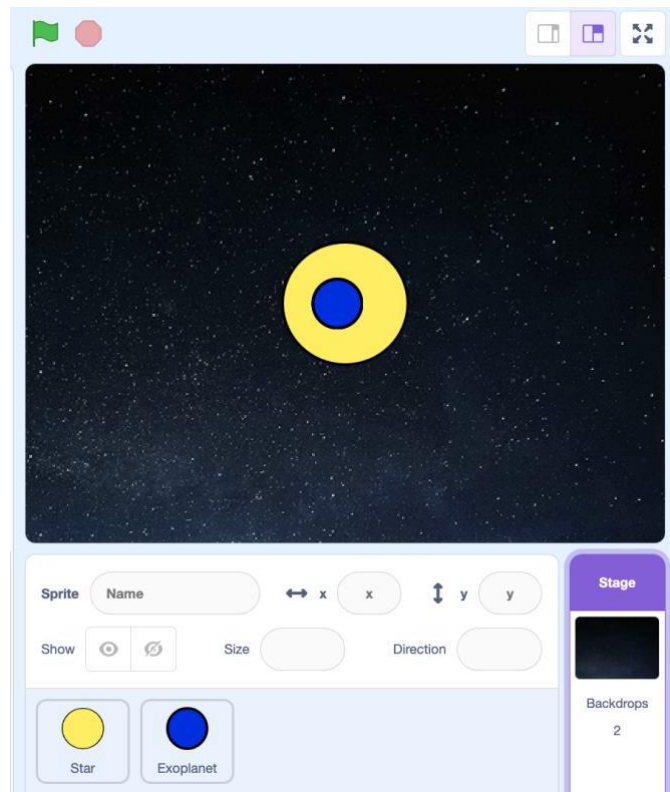
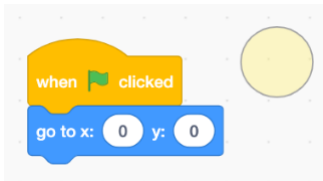
Rita landskapet som bakgrund. Skriv små program för avdunstning och regn. Fäst pilar och skriv små texter för pratbubblor.



Exoplanet i omloppsbanan

En exoplanet i omloppsbanan runt en annan stjärna än Solen.

Rita en gul stjärna och en exoplanet som två cirklar. Fastställ eller ändra storleken på exoplaneten.



Skriv ett program som får exoplaneten att "gå in i en omloppsbanan" runt stjärnan.

Astronomer har upptäckt många tusentals exoplaneter sedan 1995. En av metoderna för att upptäcka exoplaneter kallas transitmetoden.

Transitmetoden används för att se fluktuation eller attenuering i ljuset från stjärnan.

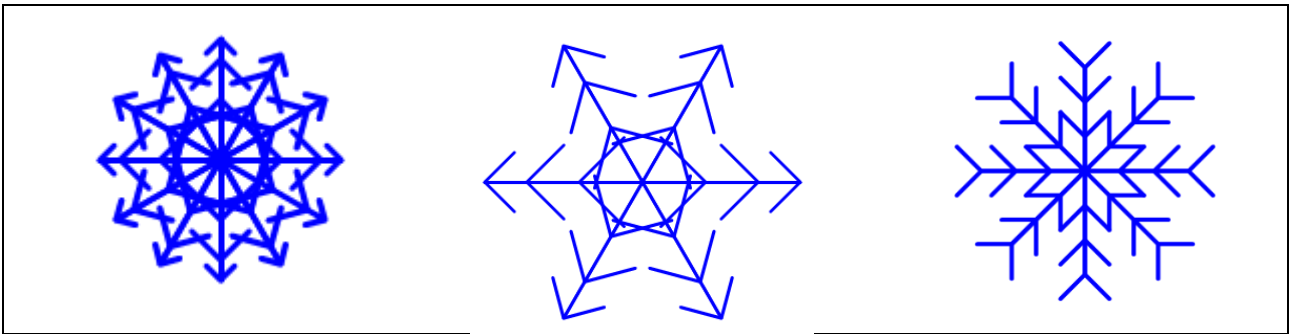
<p>The code consists of two 'when clicked' blocks. The first block contains a 'show' block, a 'forever' loop with 'go to x: -120 y: 0', 'set retning to 1', a 'repeat 240' loop with 'change x by retning', 'set retning to -1', and another 'repeat 240' loop with 'change x by retning'. The second block contains a 'forever' loop with 'show', an 'if' condition 'retning = -1 and touching Star?', and a 'hide' block.</p>	<p>Program för exoplaneten.</p> <p>Exoplaneten rör sig till höger (när riktningen är 1) och till vänster (när riktningen är -1) för alltid.</p> <p>Den andra delen av programmet kontrollerar om – så ifall exoplaneten vidrör stjärnan. När den rör stjärnan är den gömd och ser ut som att den rör sig (gömd) bakom stjärnan.</p>
---	---

Undersök:

Hur fungerar transitmetoden för exoplaneter? Hur många exoplaneter har upptäckts?

Testa det här: Visa animeringen på en stor skärm och mät ljuset med en luxmeter på en mobiltelefon.

Snökristall



Det sägs att det inte finns två likadana snökristaller i hela världen.

Upprepa tolv gånger rita samma teckning tolv gånger.

Upprepa tre gånger att rita de inverterade pilarna som blir mindre och mindre.

Variabeltalen **kontrollerar** storleken på pilarna i snökristallen.

Utmaningar

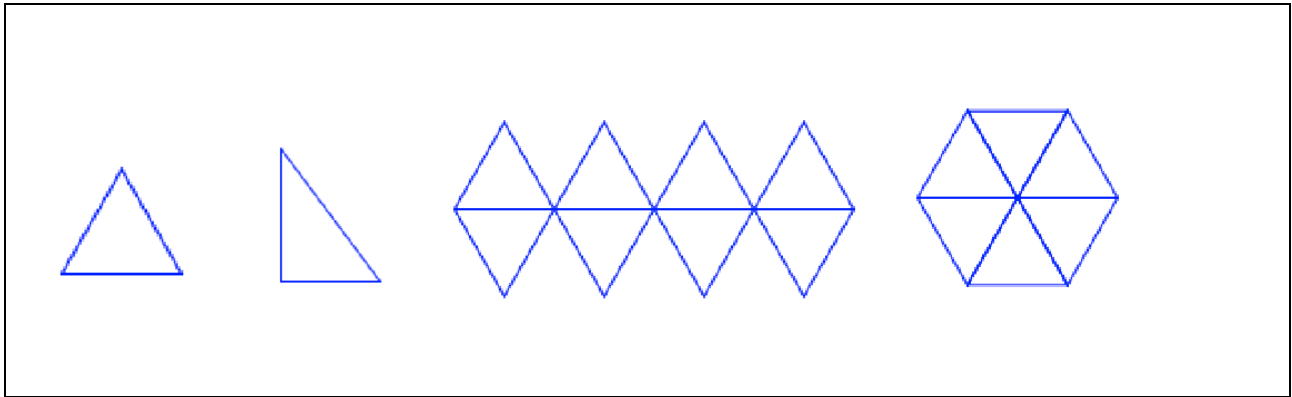
Ändra variabeltalen när du upprepar övningen och rotera graderna och rita helt nya snökristaller.

Byt färg och storlek på pennan och rita snökristaller i klara färger.

Skriv ditt eget program som ritar snökristaller.

```
when clicked
  show
  erase all
  set pen color to blue
  set pen size to 3
  go to x: 0 y: 0
  point in direction 90
  repeat 12
    go to x: 0 y: 0
    pen down
    set number to 30
    repeat 3
      move number steps
      turn 135 degrees
      move number steps
      move -1 * number steps
      turn 90 degrees
      move number steps
      move -1 * number steps
      turn 135 degrees
    change number by -10
  turn 30 degrees
  hide
```

Trianglar



Skriv koderna som ritar trianglar.

Pen down och Pen up används för att rita trianglarna.

Utmaningar
Rita dina egna trianglar.

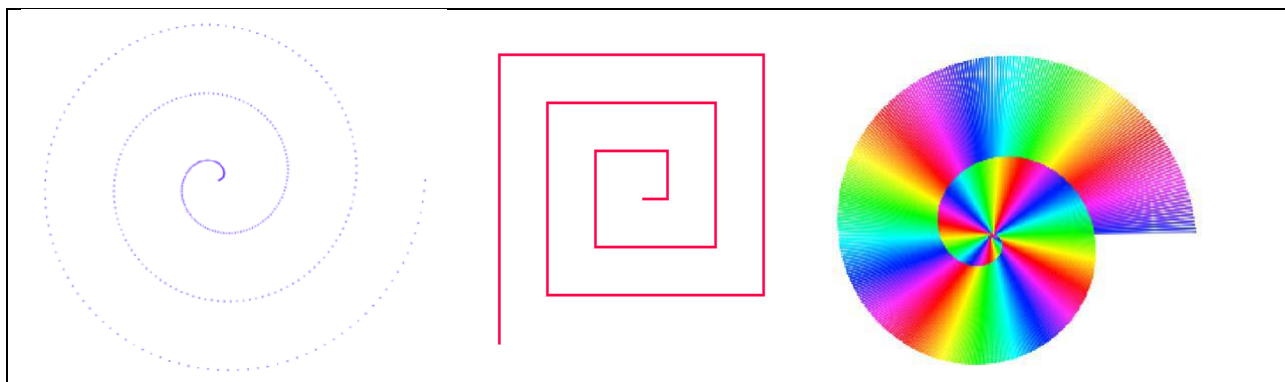
```
pen down
go to x: 0 y: 0
go to x: 60 y: 0
go to x: 0 y: 80
go to x: 0 y: 0
pen up
hide

repeat 3
  pen down
  move 50 steps
  turn 120 degrees
  pen up
turn 60 degrees
hide
```

Den här lilla koden städar upp på skärmen.

```
erase all
go to x: 0 y: 0
point in direction 90
```

Spiraler 1



Det finns två typer av spiraler.

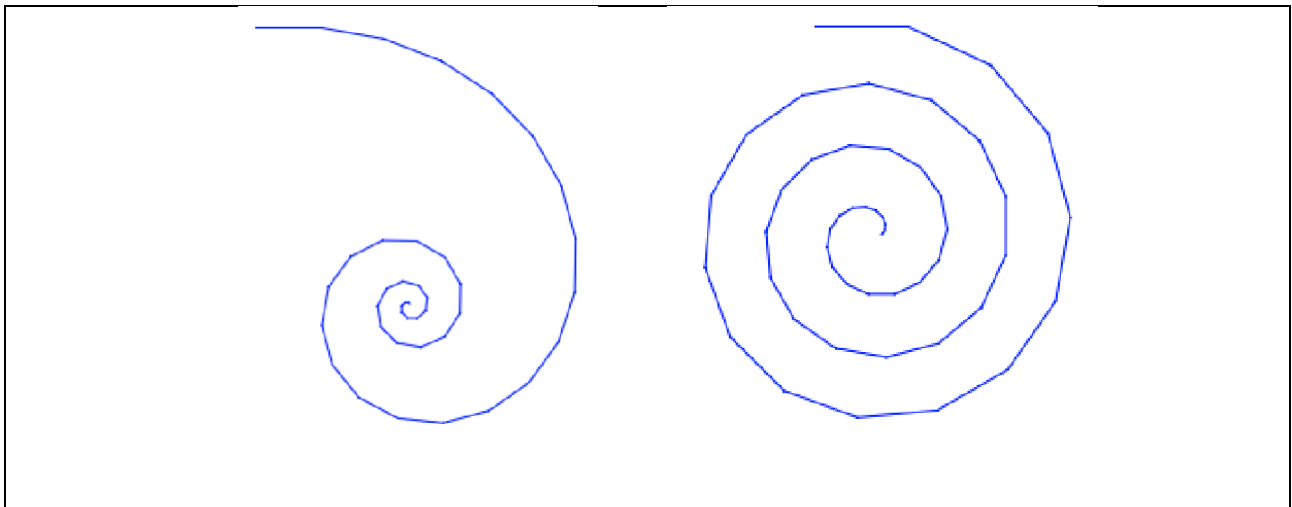
Arkimedes spiral är som en hoprullad vattenslang som har samma avstånd mellan varje varv. I en logaritmisk spiral växer avståndet mellan varven. Snäckans hus är ett exempel på en logaritmisk spiral.

Utmaning

Kan du förutsäga vilken av koderna som kommer att rita vilken av spiralerna ovan? Testa att byta ut x till ett annat tal

The image displays three Scratch code snippets on a grid background, each designed to draw a different type of spiral. The first snippet on the left uses a 'repeat' loop of 12 iterations, with 'change x by 10', 'move x steps', and 'turn 90 degrees' blocks. The middle snippet uses a 'repeat' loop of 360 iterations, with 'change x by 0.5', 'turn 3 degrees', 'move x steps', 'pen down', 'pen up', and 'go to x: 0 y: 0' blocks. The third snippet on the right uses a 'repeat' loop of 720 iterations, with 'change x by -0.2', 'turn 1 degrees', 'move x steps', 'pen down', 'change pen color by 2', and 'go to x: 0 y: 0' blocks. Each snippet is preceded by a 'show' block and followed by an 'erase all' and 'pen up' block, and the third one ends with a 'hide' block.

Spiraler 2



Det finns två typer av spiraler.

Arkimedes spiral är som en hoprullad vattenslang som har samma avstånd mellan varje varv.

I en logaritmisk spiral växer avståndet mellan varven. Snäckans hus är ett exempel på en logaritmisk spiral.

Utmaning

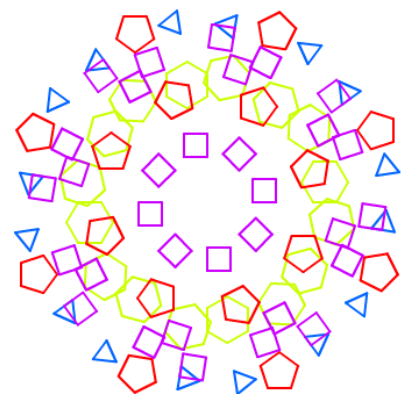
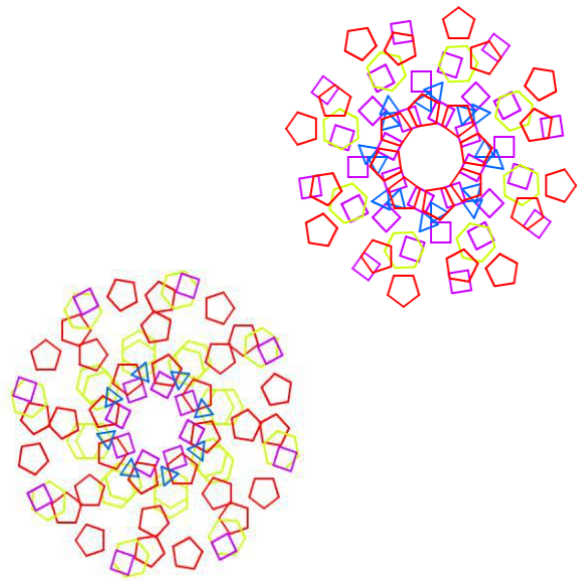
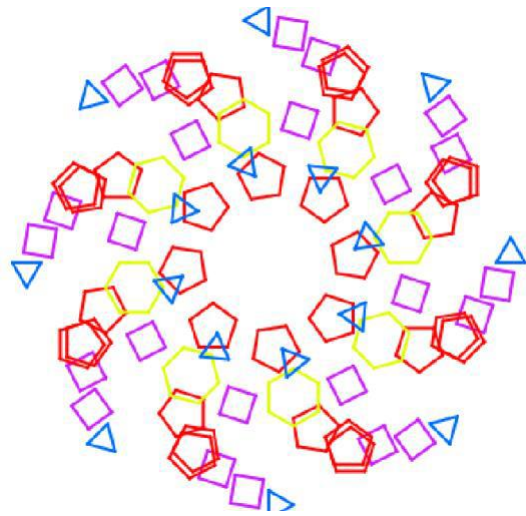


Mandala

Rita vackra mönster med polygoner i slumpmässiga färger.

Skapa två variabler "distance" och "sides in polygon".

```
erase all
show
set pen size to 2
set distance to 0
set sides in polygon to 3
point in direction 90
repeat 10
  go to x: 0 y: 0
  set distance to pick random 30 to 160
  set sides in polygon to pick random 3 to 6
  repeat 8
    turn 45 degrees
    move distance steps
    repeat sides in polygon
      set pen color to sides in polygon * 40
      pen down
      move 20 steps
      turn 360 / sides in polygon degrees
      pen up
    move -1 * distance steps
  turn 36 degrees
hide
```



BBC Micro:bit kinetisk energi hos en pendel

Micro:bit kan mäta acceleration med en inbyggd accelerometer. Den här kan vi använda när vi mäter kinetisk energi hos en svängande pendel.

Acceleration är ett tal som berättar hur snabbt hastigheten ändrar. När accelerationen är 0 finns det ingen ändring i hastigheten. Då är hastigheten konstant, alltså den samma, och ändras inte.

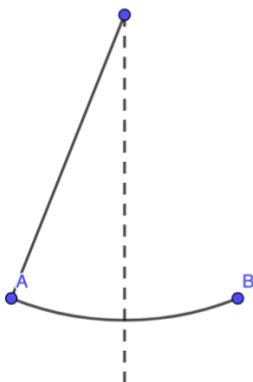
Uppgift 1:

Bygg en pendel, skriv ett program för BBC Micro:bit på <https://makecode.microbit.org> (se förslag på koden i teoridelen nedan) och använd Micro:bit för att undersöka när pendeln rör sig som snabbast (störst kinetisk energi). Micro:biten är fäst vid pendeln och utför svängningar.

Teori:

Den kinetiska energin E_{kin} är som störst när pendeln svänger förbi start-/slutpositionen. Här har änden på pendeln högst hastighet när den svänger. Kinetisk energi kallas också rörelseenergi och kan räknas ut med hjälp av en formel som innehåller hastighet.

Den potentiella energin E_{pot} är som högst i de yttersta positionerna. I de yttersta positionerna byter änden på pendeln rörelseriktning, man kan säga att änden står stilla en väldigt kort stund. Den potentiella energin kan räknas ut med hjälp av formeln $E_{pot}=m \cdot g \cdot h$, där m är massan av änden på pendeln, $g = 9.82 \text{ m/s}^2$ är jordens tyngdacceleration och h är höjden.



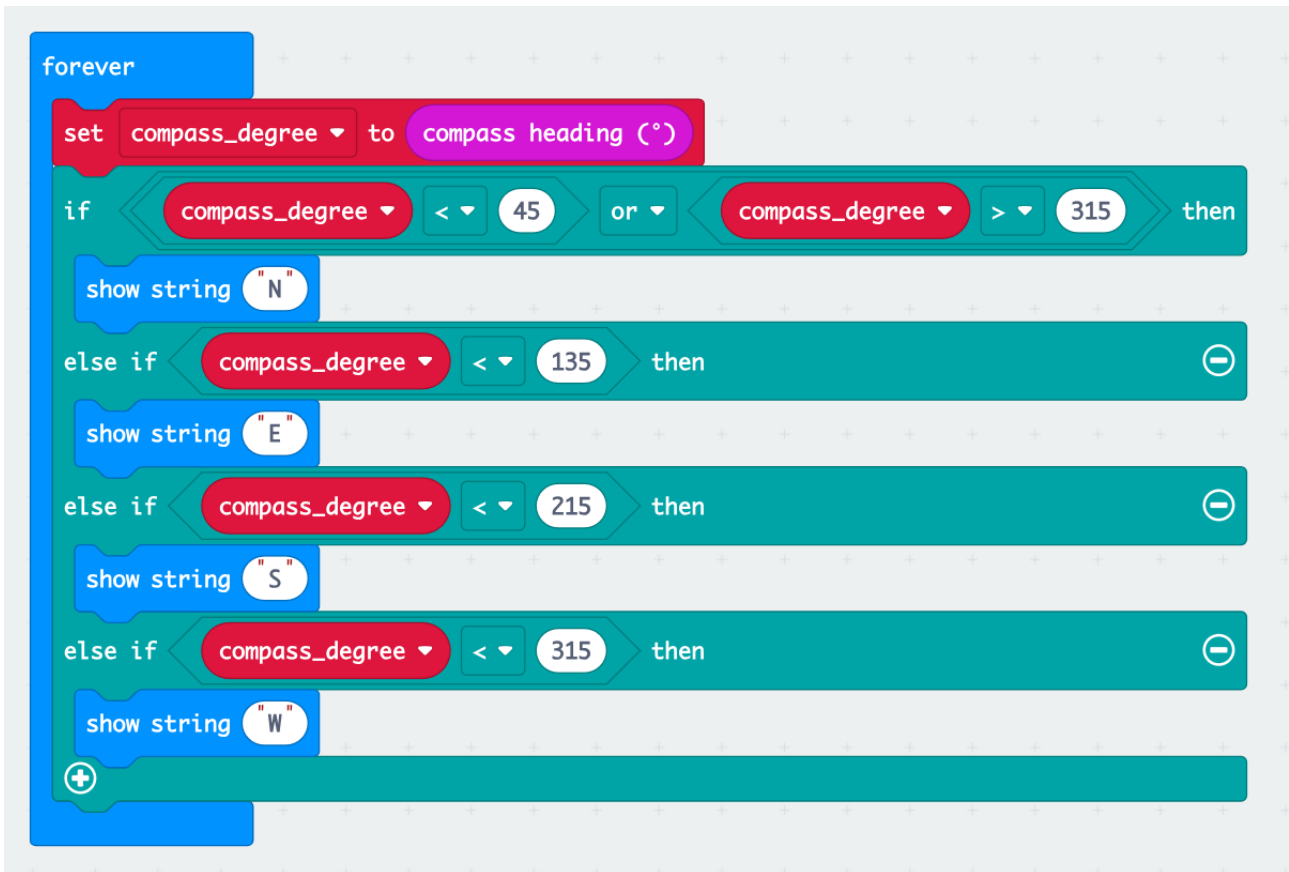
Uppgift 2:

Skriv en uppsats som är max en A4 lång där du visar och förklarar följande:

- Skriv namn/grupp, klass, ämne och datum högst upp på sidan
- Uppe på sidan, skriv: **Kinetisk energi hos en svängande pendel**
- Visa din svängande pendel och dina mätningar av kinetisk energi med hjälp av bilder.
- Förklara hur BBC Micro:bit kan mäta kinetisk energi, använd teorin och din kunskap om acceleration.
- Skapa en beskrivning av experimentet
- Skriv vilka material som används i experimentet
- Hypotes: vad tror du händer i pendelexperimentet?

Bygg en kompass med BBC Micro:bit

Bygg en kompass med BBC Micro:bit. Kompassen visar riktningarna norr, syd, öst och väst.



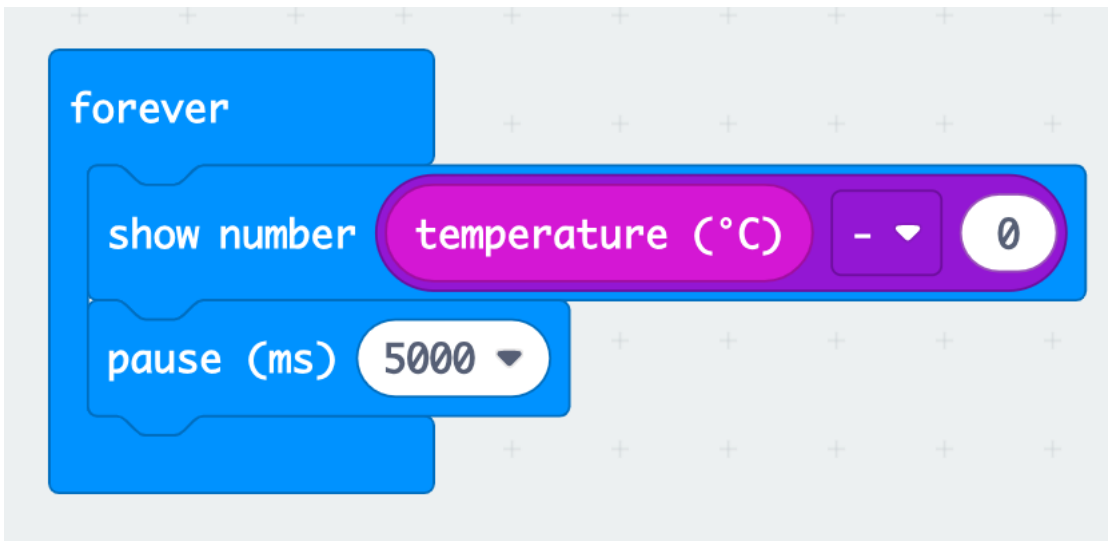
```
forever
  set compass_degree to compass heading (°)
  if compass_degree < 45 or compass_degree > 315 then
    show string "N"
  else if compass_degree < 135 then
    show string "E"
  else if compass_degree < 215 then
    show string "S"
  else if compass_degree < 315 then
    show string "W"
```

The image shows a Scratch script for a BBC Micro:bit compass. The script is contained within a 'forever' loop. It starts by setting a variable named 'compass_degree' to the value of 'compass heading (°)'. Following this, there is a series of 'if' and 'else if' statements. The first 'if' statement checks if 'compass_degree' is less than 45 or greater than 315; if true, it displays the string 'N'. The next 'else if' checks if 'compass_degree' is less than 135; if true, it displays 'E'. The third 'else if' checks if 'compass_degree' is less than 215; if true, it displays 'S'. The final 'else if' checks if 'compass_degree' is less than 315; if true, it displays 'W'. The script ends with a plus sign icon, indicating that the loop continues.

BBC Micro:bit temperatur – del 1

En Micro:bit kan mäta temperaturen i grader Celsius. I Micro:bit finns en temperatursensor inbyggd i processorn.
En processor är datorns faktiska "hjärna".

Skriv in koden så visas temperaturen. Sedan blir det en paus innan temperaturen visas igen.



Det verkar som att 0 subtraheras från temperaturen. Det låter lite mystiskt. Koden är redo för att du ska kalibrera talet som visas. Om talet är 28 och temperaturen faktiskt är 22 grader ska du subtrahera 6. Det gör du genom att skriva in 6 i det sista fältet och använda den koden i stället.

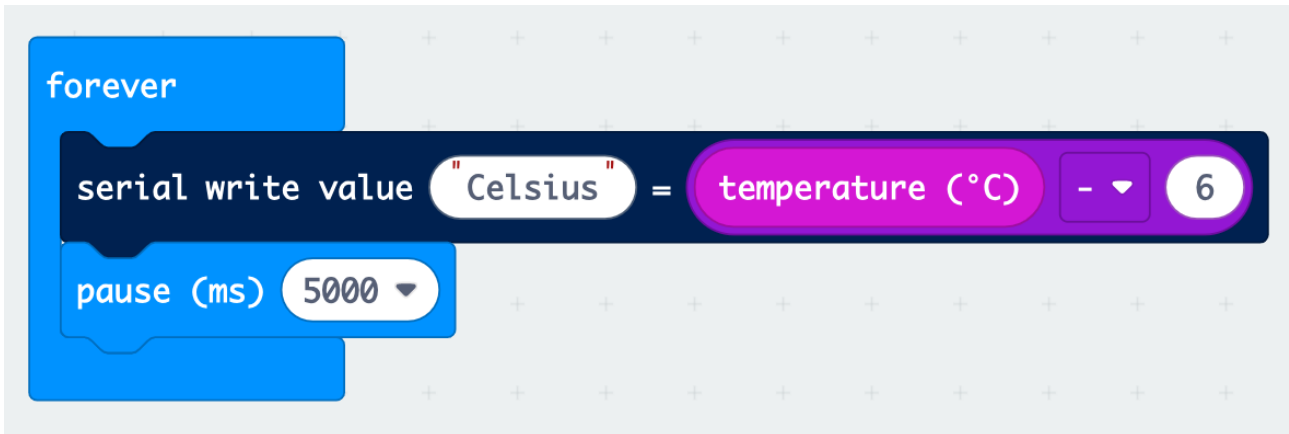


Om du vill mäta temperaturen flera gånger över tid behöver du spara mätningarna. I del två får du se hur man sparar mätningarna.
Men det kräver också att Micro:bit är kopplad till en dator.
Micro:bit kan inte spara talen själv.

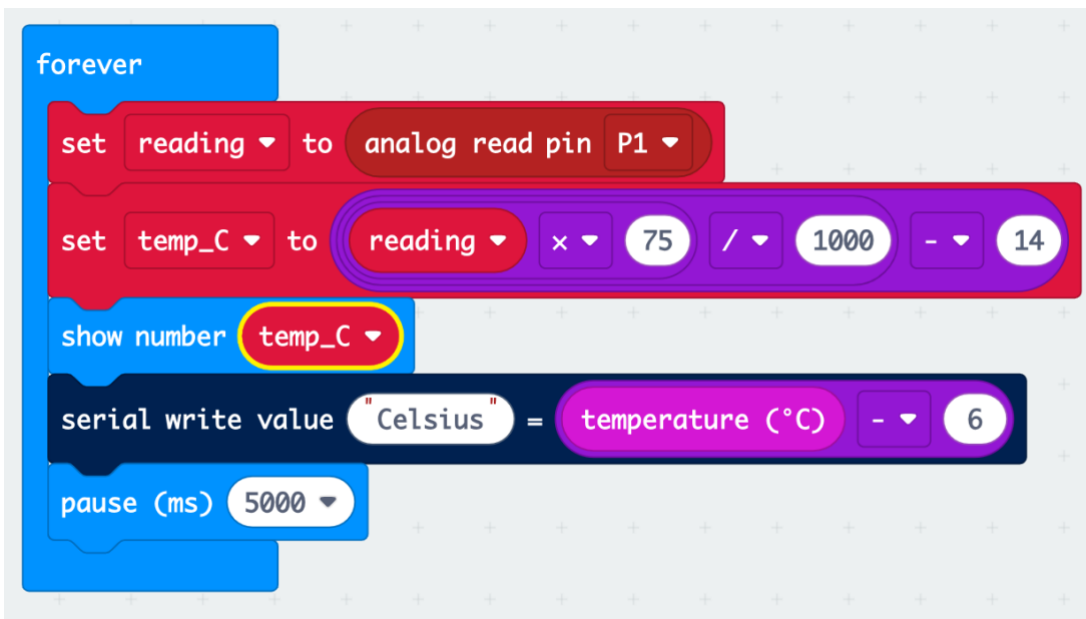
BBC Micro:bit temperatur – del 2

I Micro:bit finns en temperatursensor inbyggd i processorn. En processor är datorns faktiska "hjärna".

Jag har kalibrerat temperaturen och subtraherat 6 från talet. Det finns ett fel på 6 grader. Jag subtraherar det från temperaturen - 6.



Micro:bit och datorn är kopplade så att data visas på skärmen. Data kan registreras och laddas ner på ett kalkylark.



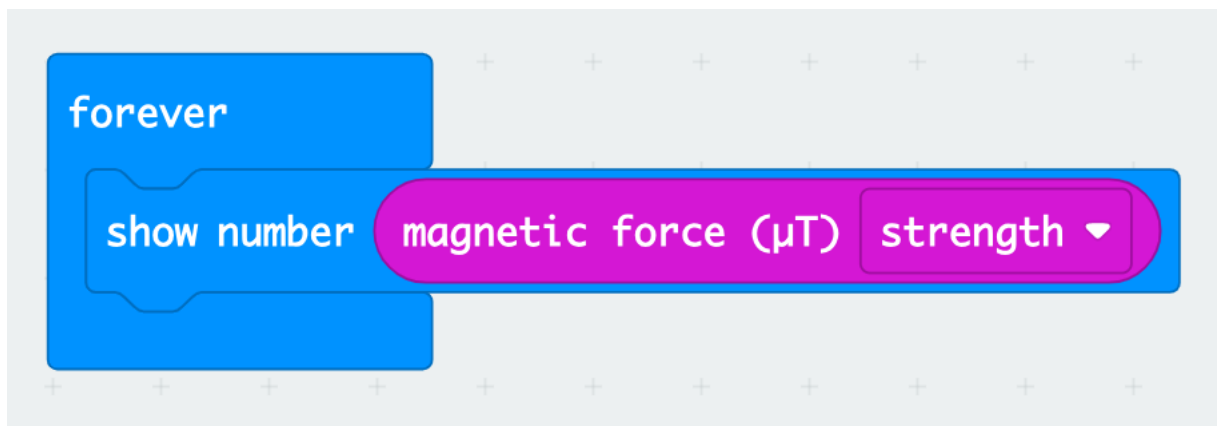
Micro:bit kopplad till dator som visar temperaturmätningarna i en graf. Data kan laddas ner i en excel-fil. Den har en liten bugg eller oregelbundenhet, eftersom formatet är 4,089 sekunder, vilket inte överförs till siffror på kalkylarket.

Källa till aktiviteten: <https://www.youtube.com/watch?v=tZy9Ev21B4c>

BBC Micro:bit magnetfältsmätare

Skriv in det här programmet i "forever"-loopen.

Micro:biten visar den magnetiska styrkan som den mäter med en liten inbyggd kompass på baksidan av Micro:biten.



Läs figuren för den magnetiska styrkan och skriv in måtten i diagrammet.

Ingen magnet	μT
En stavmagnet nära högra sidan (vid B-tangenten)	μT
En stavmagnet nära vänstra sidan (vid A-tangenten)	μT

Undersök

Vad får du reda på genom mätningarna?

Var finns kompassen på en Micro:bit?

Vad betyder μT ?