



Scratch 3.0 och matematik för årskurs 3 - 6

Kursen kommer att fokusera på vad man kan göra i Scratch (beta.scratch.mit.edu) utifrån LGR 11 och matematik i årskurs 3 – 6. Du hittar exempelfiler och dokumentation på:
<https://github.com/smeand/scratch3>

De fyra räknesätten

Tal

Avrunda

Jämna och udda tal

Enheter

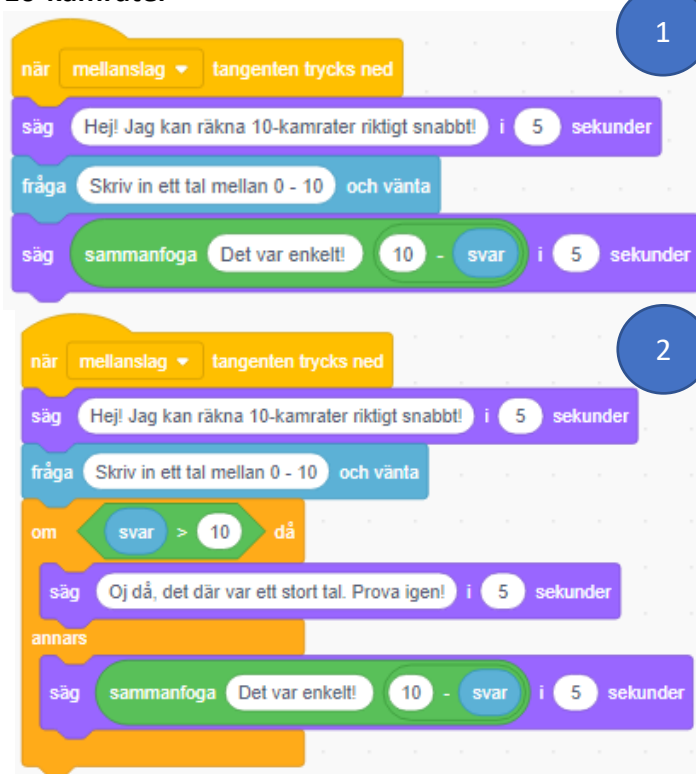
Geometri

Statistik

De fyra räknesätten

1. En katt som kan räkna snabbt

10-kamrater



Låt katten vara kvar. Pussla ihop blocken under fliken **Kod**. Tryck på mellanslag varje gång du vill göra en ny uträkning.

Det är katten som räknar och du som skriver in ett tal.

Vad händer om man skriver in ett högre tal än 10?

Det kan man åtgärda med ett villkor. Prova att utöka första koden till kod nr 2.



Scratch 3.0 och matematik för årskurs 3 - 6

Fler räknesätt

Det är bra om vi tidigt introducera variabler. Man behöver inte förstå variabler, man kan säga att man sparar talen i ett minne.

Behåll katten. Skapa två variabler, du kan kalla dem Faktor 1 och Faktor 2. Låt dem synas på scenen.

Multiplikation

Du kan skapa en ny sprajt som är ett multiplikationstecken och dra den mellan variablerna för att förtydliga mer.



1

Pussla ihop blocken som i exempel 1.



2

För att förtydliga begreppen kan du lägga till blocket **sammanfoga** och sedan skriva in **Produkten blir:** i pratbubblan.

Ändra om ovanstående kod så att den passar till addition, subtraktion och division!

Ändra värden i realtid!



3



Scratch 3.0 och matematik för årskurs 3 - 6

2. Ha koll med variabler

Behåll katten.

Skapa en variabel **Antal hopp**.

Du möter loopar i loopar för att katten ska hoppa upp och sedan ner igen.

Fundera på varför uppräkningsen av variabeln **<Antal hopp>** ligger där den ligger (kan den ligga någon annanstans?). Och varför ligger [**sätt Antal hopp till 0**] före [**för alltid**].

Vad har detta med matematik att göra?

3. Udda och jämna tal – Algoritm för att lära katten skilja på ojämna och jämna tal.


Låt katten vara kvar. Pussla ihop koden och testa att skriva in olika tal.

Operatören (funktionen) **mod** visar det som blir kvar efter en division. I det här fallet är dividenden 2 vilket gör att **mod** är perfekt för att ta reda på om ett tal är jämnt delbart med 2 eller inte.



Scratch 3.0 och matematik för årskurs 3 - 6

4. Hur snabbt kan egentligen en katt räkna?



Tid: 5.949

Tal 1 352 Tal 2 969 Resultat 341088 Tid 5.949

Låt katten vara kvar!

Skapa 4 stycken variabler.

Tal 1, Tal 2, Resultat, Tid

Låt variablerna synas på scenen, du kan dra dem i rätt ordning om du vill.

Pussla ihop blocken enligt bilden. Var noga med var du nollställer timern och var du tilldelar variabeln **Tid** timerns slutvärde.

Starta koden med mellanslag.

Blir det någon tidsskillnad om du drar bort blocket **säg[sammanfoga Resultat: <resultat>]** från repetera-loopen?

Vad är det som tar tid vid uträkningen om **säg** finns med i koden?

Varför använder vi variabeln **Tid**, räcker det inte att visa timervärdet?

```

när mellanslag ▼ tangenten trycks ned
  nollställ timer
  repetera 100
    sätt Tal 1 ▼ till slumpstal 100 till 1000
    sätt Tal 2 ▼ till slumpstal 100 till 1000
    sätt Resultat ▼ till Tal 1 * Tal 2
    säg sammanfoga Resultat: Resultat
  säg sammanfoga Tid: Tid

```

Utmana katten

Prova med olika slumpstal.

Prova olika räknesätt.

Testa med 1000 repetitioner, eller fler. Varje repetition gör en uträkning av de slumpade talen.

Med eller utan första **säg-blocket**...



Scratch 3.0 och matematik för årskurs 3 - 6

5. Leka med datum och år

Låt katten var kvar. Vi ska titta på datumfunktionen och sammanfoga text

Dagens datum



Hur många år är du?

Skapa en variabel som du kan döpa till År.





Scratch 3.0 och matematik för årskurs 3 - 6

6. Vad kan man ha för användning av en lista? Exempelvis ta reda på vilken månad det är.

Låt katten vara kvar. Skapa en lista som du kan döpa till Månad (lista finns under **Variabler**). Du behöver också skapa en variabel som du kan döpa till nummer.

Fyll listan med namnen på månaderna. Du ska ha 12 poster med månadsnamn i listan.

Funktionen **<aktuell månad>** ger dig månadsnummer ex. 1 för januari. Du stegar dig igenom listan tills du har hittat rätt månadsnummer och visar den posten från listan.

Variabeln **nummer** används bara för att kunna stega igenom listan.

Försök att lista ut hur koden fungerar. Skulle du kunna använda engelska och svenska i samma lista?

Hur ska man tänka då? Eller vilket stjärntecknet är för de som är födda i den här månaden.

The image shows a Scratch workspace with a stage and a script area. On the stage, there is a list titled "Månad" with 12 items: Januari, Februari, Mars, April, Maj, Juni, Juli, Augusti, September, Oktober, November, and December. The list has a scroll bar and a "length 12" indicator. To the right of the list, there are two variables: "nummer" with a value of 10 and "Månad" with a value of "Oktober". A speech bubble above the cat says "Oktober". In the script area, there is a code block that starts with "när flagga klickas på" (when green flag clicked). The code block contains the following steps: "sätt nummer till 0" (set nummer to 0), "sätt Månad till " (set Månad to), "repetera" (repeat) with "aktuell månad" (current month) and "år" (year) as arguments. Inside the repeat loop, there are three steps: "ändra nummer med 1" (change nummer by 1), "sätt Månad till post nummer i Månad" (set Månad to the value at index nummer in the Månad list), and "säg Månad" (say Månad for 2 seconds).

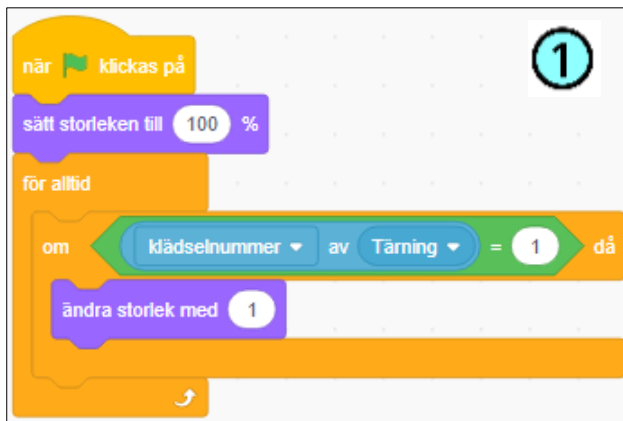
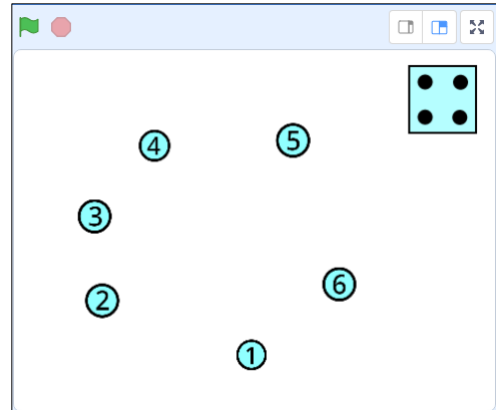


Scratch 3.0 och matematik för årskurs 3 - 6

7. Tärningen är kastad, simulering med slumptal och statistik

Ta bort katten. Rita en sprajt (döp till tärning) med 6 olika klädselar (med tärningens olika värden på) som du döper till 1, 2, 3, 4, 5, 6. Glöm inte att du kopierar klädseln och förändrar den med fler och fler prickar (kopiera pricken och klistra in den för att utöka antalet). Koda enligt bilden till höger.

Nu ska du rita en sprajt (cirkel) med siffran 1. Lägg in nedanstående kod. Kopiera sedan sprajten så att koden följer med och gör sedan förändring i nya sprajts klädsel och även i koden så att det stämmer med siffran. Kopiera så att du har sex cirkelar med förändrad kod och siffra i sprajten.



Placera ut cirkelarna på scenen och starta allt med Flaggan. Det som ska ske är att tärningen simulerar kast (500 st) med 0.1 sek mellanrum och varje gång klädselnumret på tärningen stämmer med siffran i cirkeln så ändras cirkelns storlek. Vilken sprajt blir störst, dvs. vilken siffra kommer upp flest gånger vid simuleringen?

Man kan diskutera om slumpalen verkligen slumpar fram varje gång. Om man har gjort om simuleringen tillräckligt många gånger bör alla siffror ha "vunnit".

Utmaning

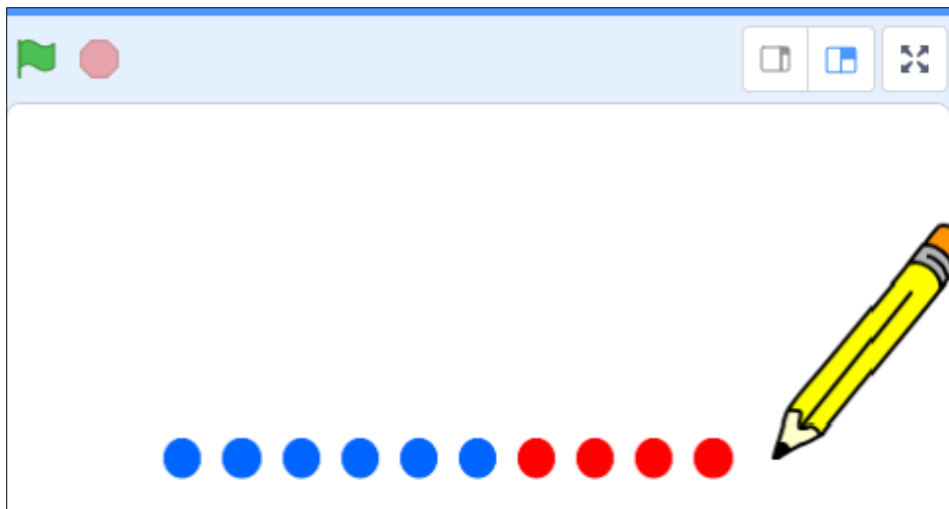
Hur skulle du ha gjort om du vill ha exakta antalet (1:or resp. 6:or) för alla tärningskastet?

Variabler, kan kanske vara någonting? Testa!



Scratch 3.0 och matematik för årskurs 3 - 6

8. Rita på scenen – visualisera matematik – 10 kamrater



Ta bort katten. Hämta en sprajt (pencil) och se till så att spetsen på pennan är vid centrummärket i fliken klädselar.

Hämta tillägget **Penna** (ikonen längst ned till vänster).

Sätt upp alla förutsättningar först. Radera allt, pennans storlek, färg och startposition.

Om man sätter blocken **penna ned** och direkt efter **penna upp** så får man en dutt (en punkt) på scenen.

Först slumpar vi fram ett 1 - 10 dattar med 30 punkters mellanrum (+x).

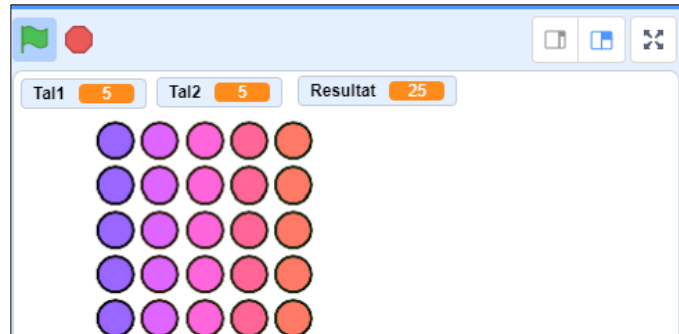
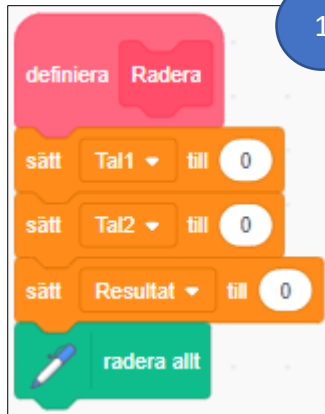
Därefter frågar vi hur många dattar det är kvar till 10. Vi ändrar färgen till röd på duttarna.

Om vi säger fel har vi inget som rättar oss. Kan du fixa det?



Scratch 3.0 och matematik för årskurs 3 - 6

9. Multiplikation med visuellt resultat



Ta bort katten. Rita en mindre **sprajt** (fylld cirkel)

Skap 3 variabler; **Tal1**, **Tal2**, **Resultat**.

Hämta tillägget **Penna**.

Skap två **funktioner** under **Mina Block**.

1. Radera. Nollställer variablerna och **raderar allt**.

2. ShowDots med indata Tal1 och Tal2 (ljudet kan du lägga till om du vill)

3. Programmera huvudprogrammet. Testa!





Scratch 3.0 och matematik för årskurs 3 - 6

10. Scenen och koordinatsystemet

1. Rita i ett koordinatsystem

Byt ut katten mot en penna. Sätt centrum på pennan till spetsen (under klädslar).

Leta upp bakgrunden som heter XY-grid och lägg till den.

Välj **tillägget Penna** (ikonen längst ned till vänster) så att du får blocken för pennan.



Lägg in de här blocken i fliken Kod för sprajten penna.

Starta koden med den gröna flaggan. Koden utförs sekventiellt och avslutas med att pennan tas upp igen.

Prova sedan att rita upp följande koordinater:

Start $x = 0$ och $y = 0$,

glid till koordinaterna

$x = -100$, $y = 100$

$x = -100$, $y = 0$

$x = 0$, $y = 0$

Kan du räkna ut

vilken geometrisk figur det blir?

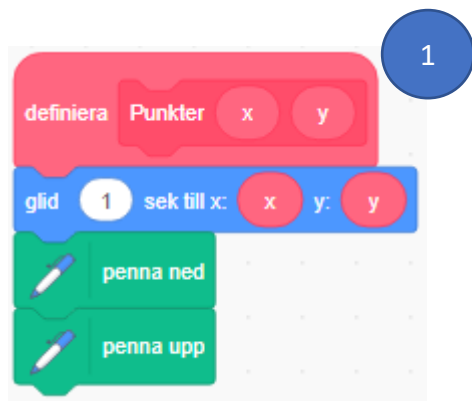
Rita en till i nedre högra fältet i koordinatsystemet.

Du får lägga till fler glida-block för att lösa uppgiften.

2. Sätta ut punkter i koordinatsystemet.

Lägg först till tillägget **Penna** om du inte redan har det. Och låt Xy-grid ligga kvar som bakgrund och sprajten penna som i tidigare exempel (centrum på pennan ska vara vid spetsen).

Nu ska vi lära oss att skapa en **funktion** (ett eget block – **Mina block**). Skapa ett block och döp det till **Punkter** och lägg till två indata, X och Y. Livet kommer att bli mycket enklare efter detta! Koda enl. bild 2. Kör!





Scratch 3.0 och matematik för årskurs 3 - 6

11. Multiplikationstabellen (lite överkurs)

The screenshot shows the Scratch 3.0 interface. The code area contains the following script:

```
when green flag clicked  
  when green flag clicked, tangent is pressed down  
  go to x: -172 y: 128  
  clear all  
  ask "Enter a factor (1 - 10)" and wait  
  set Faktör 1 to answer  
  ask "Enter a factor to multiply (1 - 10)" and wait  
  set Faktör 2 to answer  
  pen down  
  repeat (Faktör 1 - 1) times  
    change x by 43  
  repeat (Faktör 2 - 1) times  
    change y by -32.1  
  pen up
```

The multiplication table is displayed on the right, with the value 54 highlighted in the cell for 9x6.

Importera bakgrunden (tabellen) som finns på githubsidan.

Skapa två variabler Faktör 1 och Faktör 2.

Ta bort katten och skapa en röd rektangel (röd kant runt bara) som täcker en ruta i tabellen.

Glöm inte att välja tillägget penna!

Koda rektangeln med koden till vänster. Testa med ändra x och ändra y så att de stämmer med din rektangel.

Fundera på varför repetera-loopen minskas med 1 (Faktör 1 - 1). Vad händer om det inte står -1?

Stämmer uträkningen?



Scratch 3.0 och matematik för årskurs 3 - 6

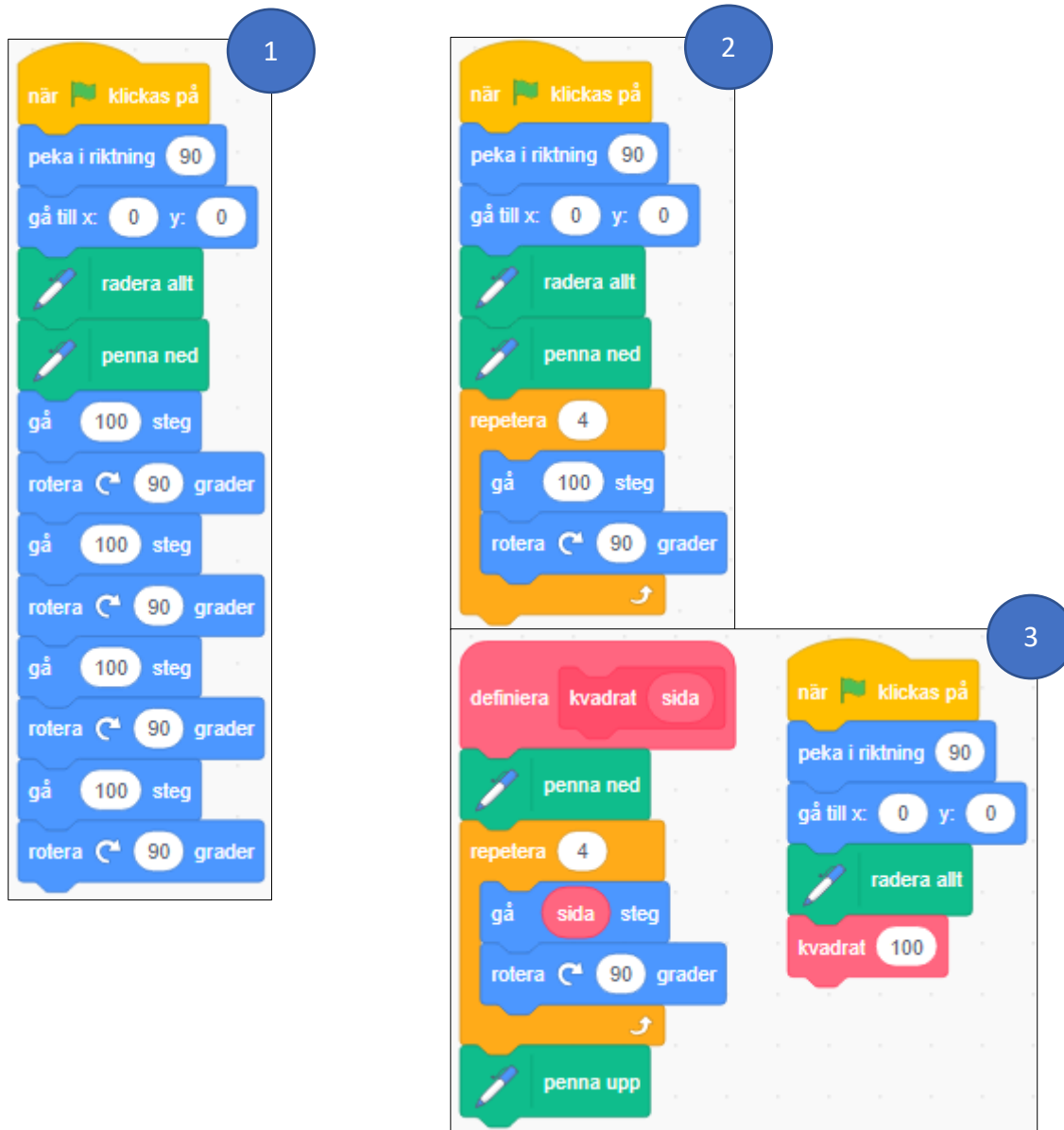
12. Geometriska figurer på olika sätt

Rita en kvadrat och förenkla algoritmen

Ta bort katten. Rita en ny sprajt som ett litet plustecken + eller bara en punkt. Det är den sprajten som ska rita upp olika geometriska figurer. Lägg till tillägget **pennan** så att du kan rita.

Vi ska förenkla och förbättra en algoritm i flera steg. Vi ska rita en kvadrat med sidan 100 pixlar.

1. Så här kan vi lösa problemet med sekventiell kod.
2. Vi ser ett mönster som kan förenklas med en loop.
3. Vi kan skapa en funktion som vi kan förändra och ropa på flera gånger. OBS! Bra att lägga in **[penna upp]** och **[penna ned]** i funktionen kvadrat. På så sätt kan du rita kvadrater var som helst utan att förbinda dem med streck.



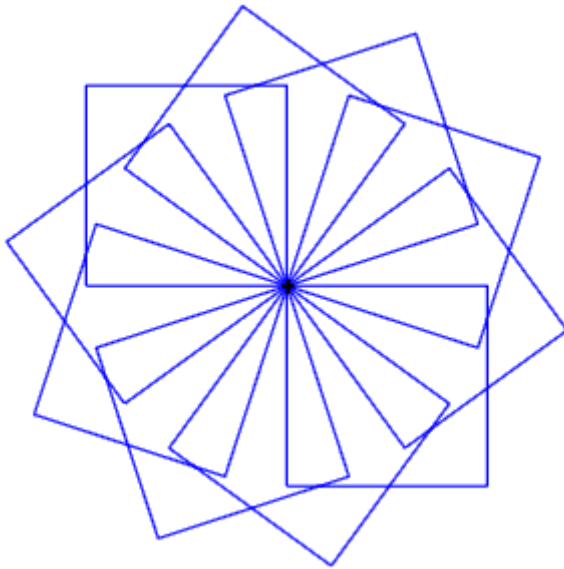


Scratch 3.0 och matematik för årskurs 3 - 6

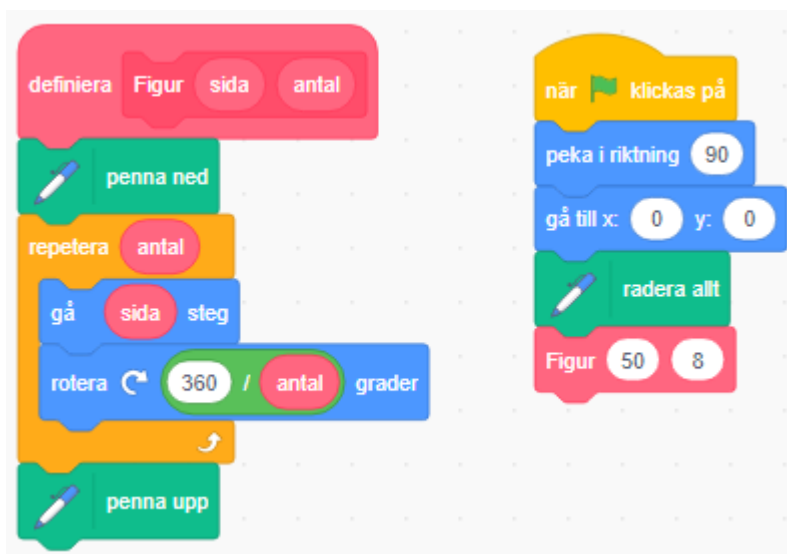
Flera kvadrater

Lite utmaningar

1. Använd din senaste funktion för en kvadrat. Försök koda så att 10 stycken kvadrater ritas upp på slumpmässiga positioner på scenen. Leta efter blocket som ger en slumpmässig position.
2. Kan du tänka ut hur den här figuren är skapad? Använd din funktion kvadrat. Tips! Ett varv är 360 grader. Skissa gärna på papper om du vill hitta algoritmen.



3. Förändra funktionen Kvadrat så att du enkelt kan rita olika geometriska figurer. Bra att döpa om den till exempelvis Figur istället för kvadrat. Fundera lite på hur funktionen är upplagd. Namnet – **Figur** och även **två indatafält** vilket betyder att vi kan förändra två variabler i funktionen. Prova att skapa ovanstående figur fast med oktogoner i stället för kvadrater. Vad är tricket?

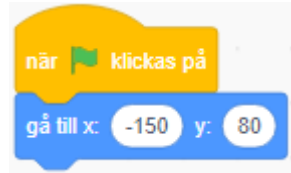




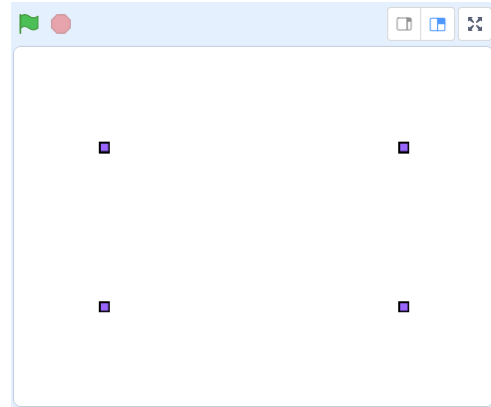
Scratch 3.0 och matematik för årskurs 3 - 6

13. Teknik för att rita geometriska figurer i realtid

Ta bort katten. Skapa en sprajt som du kan kalla **nod1**, helst en liten fylld cirkel eller kvadrat som är centrerad. Placera den uppe till vänster, sedan lägger du in startblocket med positionen. När det är klart kopierar du sprajten 3 gånger så att du har fyra noder **nod1-nod4**.



Lägg till de nya positionerna så att de bildar en rektangel, dvs. nod2 kan ha positionen x=150 och y=80 osv...



Skapa en sprajt till (kalla den **penna**), som är en liten (3-4 px) centrerad punkt.

Nu är det dags för den tekniska lösningen för realtidsanimeringen. Ganska enkel egentligen.

Koda sprajten **penna**!



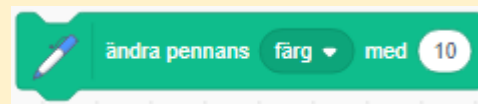
Du måste lägga till tillägget **Penna** så att du hittar pennans block.

Fundera lite på vad sprajten penna kommer att göra i koden till vänster.

Starta programmet och flytta på en av noderna. När du släpper noden kommer en ny geometrisk figur att ritas upp.

Idéer som du kan testa

- Vad händer om du flyttar blocket [radera allt] efter nod4? Fundera lite varför det blir som det blir.
- Vad händer om du flyttar ut samma block utanför [för alltid] exempelvis direkt under [när gröna flaggan klickas på]? Testa att dra mer än en nod, vad händer? Kan du göra snygg geometrisk figur?
- Prova att lägga in nedanstående block under **nod4**.



- Vad kommer att hända om du väljer att utöka koden (se bilden) under nod1?
- Testa att utöka koden för de andra noderna också.





Scratch 3.0 och matematik för årskurs 3 - 6

14. Överkurs med realtidsfiguren

Idén är att vi ska kunna räkna ut omkrets och area på figuren i realtid. Det enklaste är att vi enbart håller oss till rektangeln alternativt kvadraten. Alltså om vi ändrar nod1 ska också nod2 och nod4 följa med.



nod1

1. Börja med att göra **nod1** dragbar (fungerar vid scenen i fullskärmsläge)
2. Utöka **nod2** med koden så att **nod2** följer nod1:s y läge, dvs. nod1 och nod2 följs åt i y-led.
3. Gör samma sak för **nod4** så att den följer nod1:s x-läge.



nod2

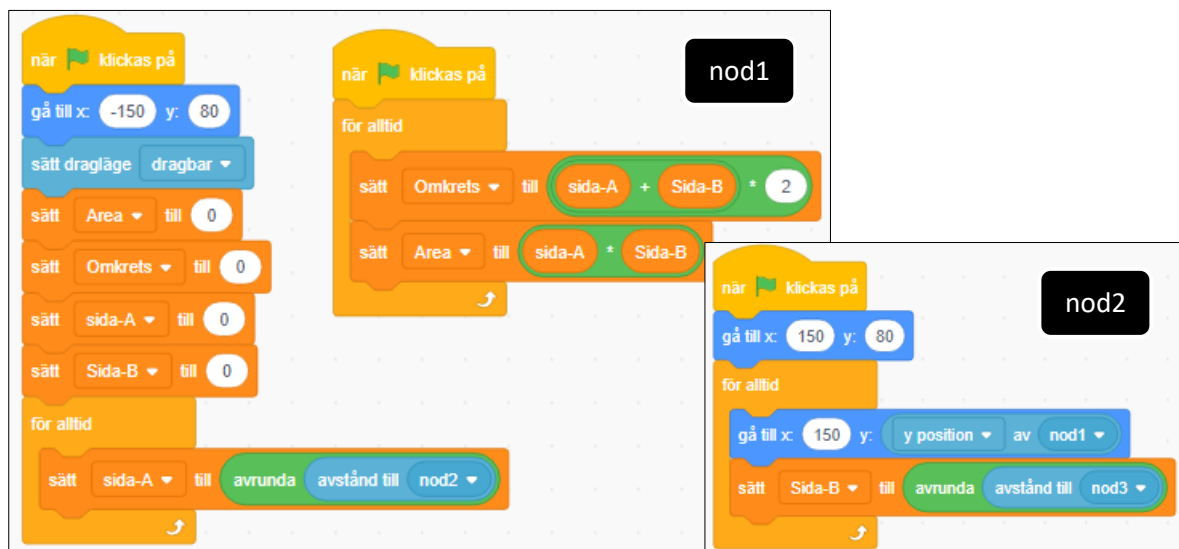


nod4

4. Testa så att allt fungerar i fullskärmsläge, du ska då enbart kunna dra nod 1, men nod2 och nod4 följer med.

5. Dags för lite realtidsuträkning av omkrets och area.

- Skapa fyra variabler: Area, Omkrets, sida-A, sida-B. Låt dem vara synliga på scenen.
- Mäta avstånd. Gå till nod1:s kod och sätt dit alla variabler och nollställ dem innan **[för alltid]**
- Sätt variabeln **sida-A** till **avrunda[avstånd till nod2]** i **nod1:s** kod
- Sätt variabeln **sida-B** till **avrunda[avstånd till nod3]** i **nod2:s** kod
- Starta programmet och se så att värdet på **sida-A** och **sida-B** förändras i realtid när du ändrar rektangeln via **nod1**. Lägg in uträkningen av omkrets och area i en parallell process i nod1.





Scratch 3.0 och matematik för årskurs 3 - 6

15. Skanna en okänd rektangel och räkna ut area och omkrets (överkurs)

Ta bort katten. Skapa 4 variabler; Sida-A, Sida-B, Omkrets, Area.

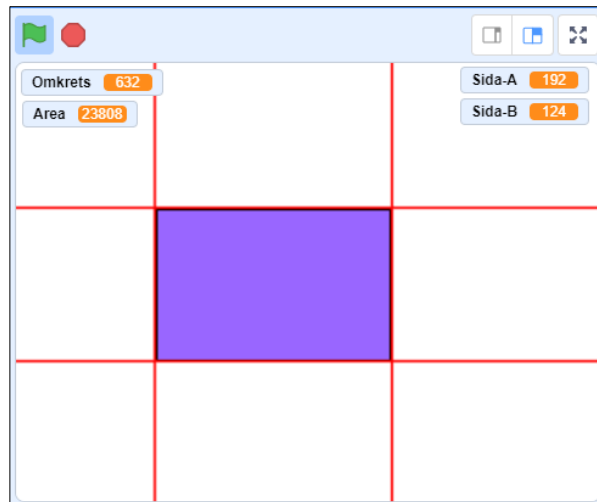
Rita en sprajt som en rektangel med svart kant. Kopiera flera klädselar och ändra storleken så att det finns några olika att välja mellan (pil upp).

Rita en sprajt till, ett horisontellt rött streck ungefär 3-5 pixlar brett (döp till h1).

Skriv in koden för sprajten h1 (bild 1)

Kopiera sprajten h1 (blir automatiskt h2)
Förändra koden så att strecket går uppåt från y-positionen -180.

Kopiera igen och döp till v1 (ändra klädseln att det blir ett vertikalt streck) och förändra koden så att strecket rör sig åt höger från x-position -240. Kopiera sprajten v1 till v2 och förändra koden så att strecket går åt vänster från x-position 240.



så
1



2

Lägg in den här koden (bild 2) i sprajten med rektanglarna.

Växla klädsel med **uppåtpil**, och starta skanningen med **mellanslag**.

Det är bra att avrunda de skannade positionerna. **abs()** ger ett positivt värde för alla koordinaterna, vilket gör allt lite lättare.

when clicked:

- change costume to: costume1
- go to x: 0, y: 0
- set Omkrets to: 0
- set Area to: 0
- set Sida-A to: 0
- set Sida-B to: 0

for all:

- set Sida-A to: round(abs(x position of v1) + abs(x position of v2))
- set Sida-B to: round(abs(y position of h1) + abs(y position of h2))
- set Omkrets to: Sida-A + Sida-B * 2
- set Area to: Sida-A * Sida-B