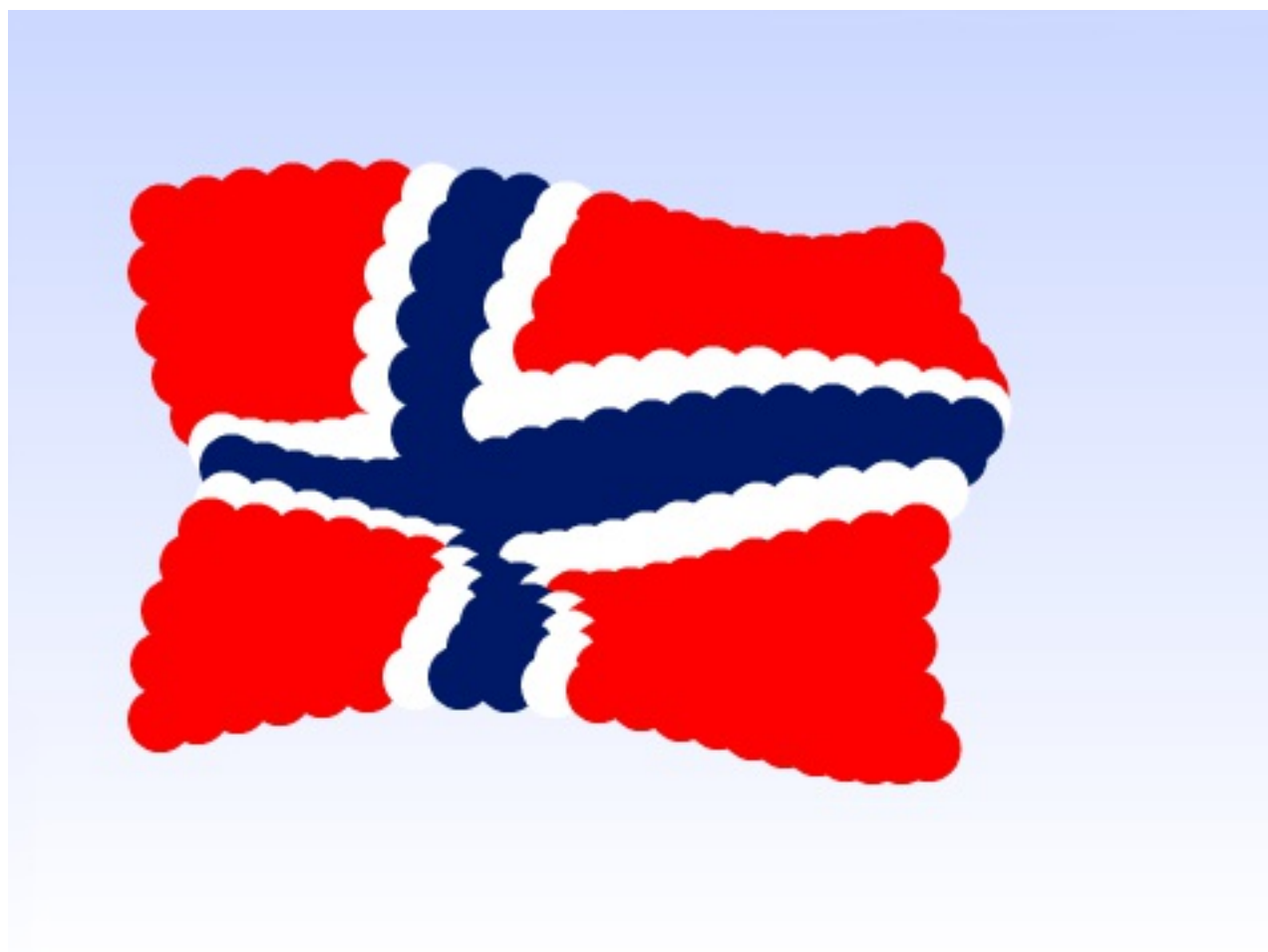




## Introduksjon

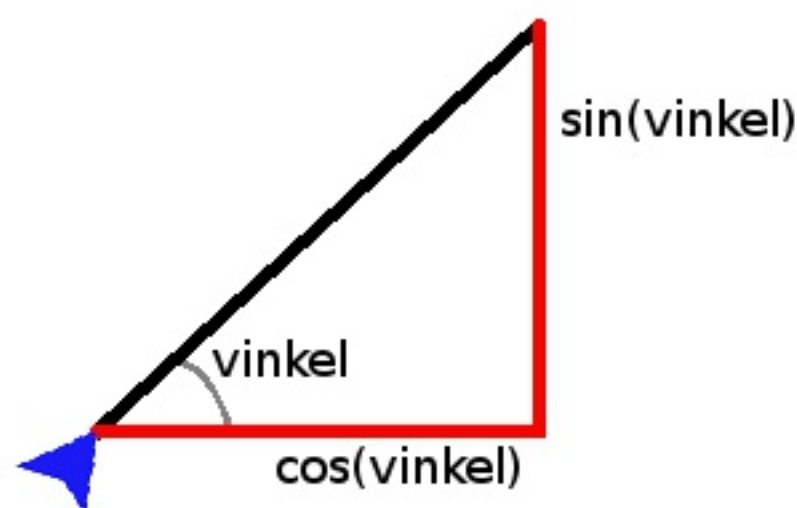
I denne oppgaven vil vi se litt nærmere på hvordan vi kan lage spennende mønstre og animasjoner ved hjelp av litt matematikk. Spesielt skal vi tegne et flagg som ser ut som om det vaier i vinden.



## Steg 1: Gå i sirkel

Vi har tidligere sett flere måter vi kan få figurer til å gå i sirkel på. For eksempel flyttet vi i [Soloball](#) figuren vekk fra sentrum og brukte [pek mot musepeker](#) for å flytte figuren i sirkel.

I denne oppgaven skal vi bruke to funksjoner, *sinus* og *cosinus*, for å ta litt mer kontroll over hvordan sirkelbevegelsen skjer. Du har kanskje ikke hørt om *sinus* og *cosinus*? Disse er tall som forteller hvor lange sidene i en rettvinklet trekant er i forhold til hverandre. I figuren nedenfor forteller for eksempel *cosinus*, `cos(vinkel)` oss hvor lang den vannrette streken er i forhold til den skrå streken.



La oss se hvordan vi kan bruke dette til å tegne sirkler!



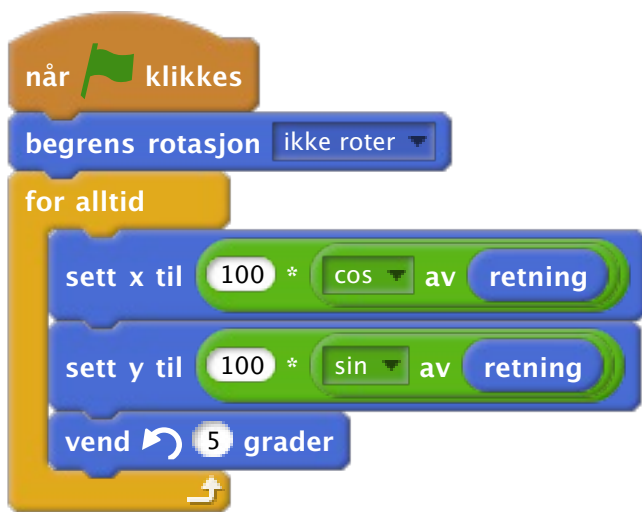
## Sjekkliste



Start et nytt prosjekt.



Skriv det følgende skriptet på kattefiguren:



## Test prosjektet

Klikk på det grønne flagget.

- ☐ Hva skjer? Skjønner du hvorfor kattefiguren flytter seg rundt i en sirkel?  
Tallet **100** forteller hvor stor sirkelen skal være (det er lengden av den skrå streken i figuren over). Prøv å endre på tallet (begge steder) for å se hva som skjer.
- ☐ Hva skjer om du bruker forskjellige tall de to stedene i stedet for **100** ?
- ☐ Hvordan forandrer bevegelsen til katten seg om du forandrer tallet **5** i **vend 5 grader** ? Prøv gjerne med negative tall også!

## Retning på sirkelbevegelsen

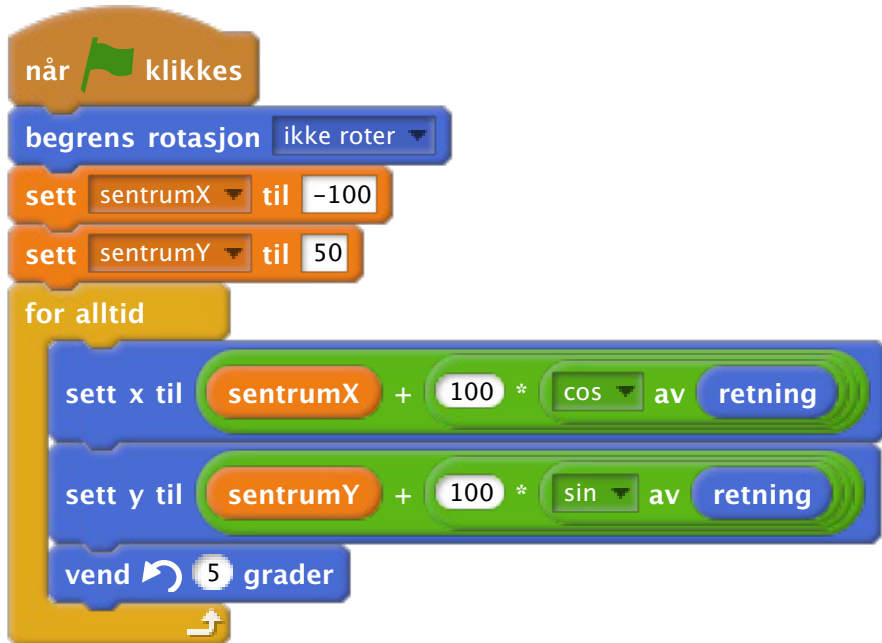
Du legger kanskje merke til at katten flytter seg i motsatt retning av hvordan pilen i **vend 5 grader** klossen peker? Dette er fordi Scratch måler vinkler i motsatt retning av hva som er vanlig i matematikken (og som vi har tegnet over).

# Steg 2: Flytt sirkelen

Så langt har vi bare tegnet en sirkel midt på skjermen. La oss se om vi kan flytte den!

## Sjekkliste

- ☐ Siden vi allerede flytter figuren langs en sirkel kan vi ikke bruke **gå til x: y:** -klosser for å flytte hele sirkelen. I stedet vil vi bruke variabler. Lag to variabler som heter **sentrumX** og **sentrumY** og som gjelder kun *for denne figuren*.
- ☐ Vi kan nå flytte sirkelen over ved å sette **sentrumX** - og **sentrumY** -variablene:



# Test prosjektet

Klikk på det grønne flagget.

☐ Flytter sirkelen seg til det nye sentrumet?

## Prøv selv

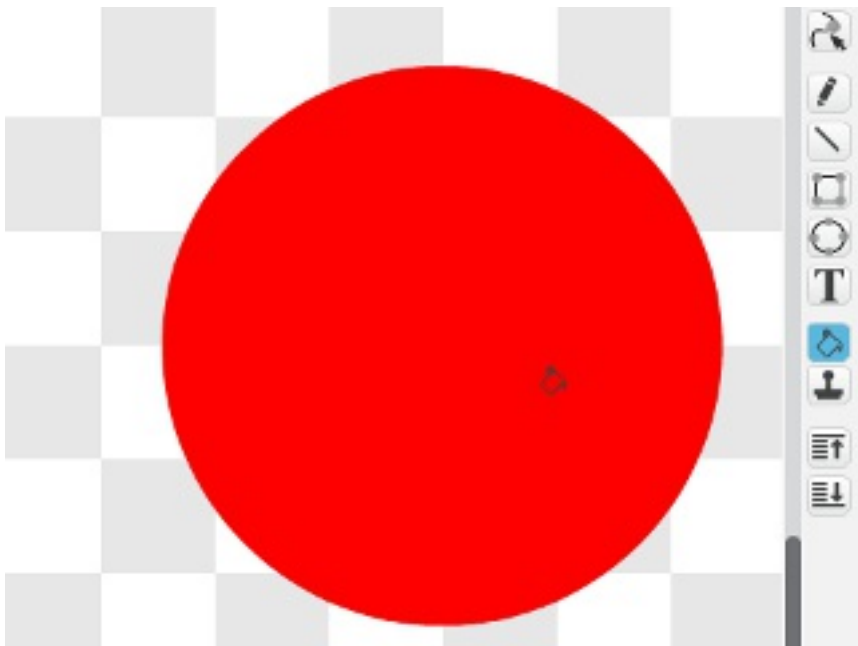
- ☐ Legg til en ny variabel, `radius`, som også gjelder kun *for denne figuren*. Kan du bruke denne til å styre hvor stor sirkelen er? Det vil si, `radius` skal fortelle lengden av den skrå streken i figuren i begynnelsen av oppgaven.
- Du trenger en `sett radius til` -kloss i tillegg til å bruke `radius` to steder i koden din.

## Steg 3: Dansende disker

Nå skal vi prøve å få mange figurer til å gå i sirkel samtidig.

### ✓ Sjekkliste

- ☐ Tegn en ny drakt på figuren din. Bruk vektorgrafikk til å tegne en rød fylt sirkel (disk). Denne kan være ganske liten, for eksempel `20 x 20` piksler.







- ☐ Vi kan lage mange røde disker ved å klonе denne ene disken. Del koden din i to deler, og endre den på denne måten:

```

når flagg klikkes
  gjenta 99 ganger
    sett sentrumX til tilfeldig tall fra -150 til 150
    sett sentrumY til tilfeldig tall fra -100 til 100
    sett radius til 50
    lag klon av meg

når jeg starter som klon
  for alltid
    sett x til sentrumX + radius * cos av retning
    sett y til sentrumY + radius * sin av retning
    vend 5 grader
```

- ☐ Om du kjører programmet ditt nå vil du se 99 disker som danser (og 1 som står i ro) rundt på skjermen i et herlig kaos! Kaoset kommer faktisk av at Scratch bruker litt tid på å starte hver klon. Vi kan lage en mindre kaotisk dans på denne måten:

Bytt  med , og legg til en  -kloss etter  -løkken.

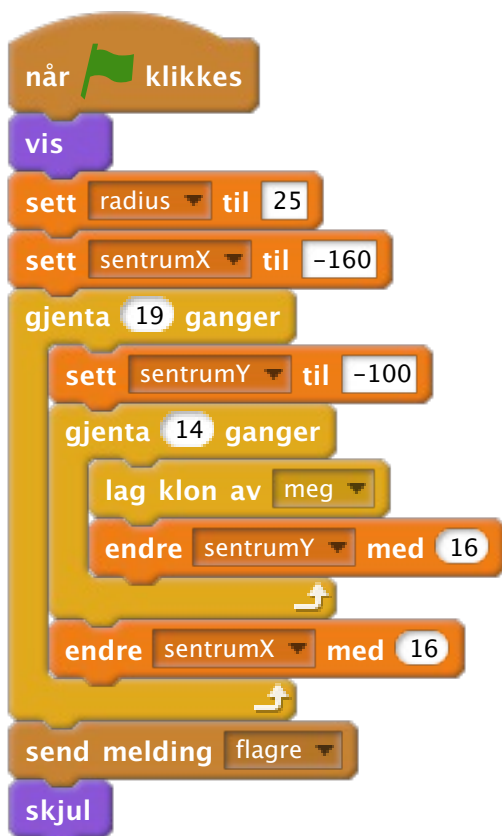
Om du kjører programmet ditt igjen vil du se at alle diskene danser i takt. Hva liker du best?

## Steg 4: Flagrende flagg

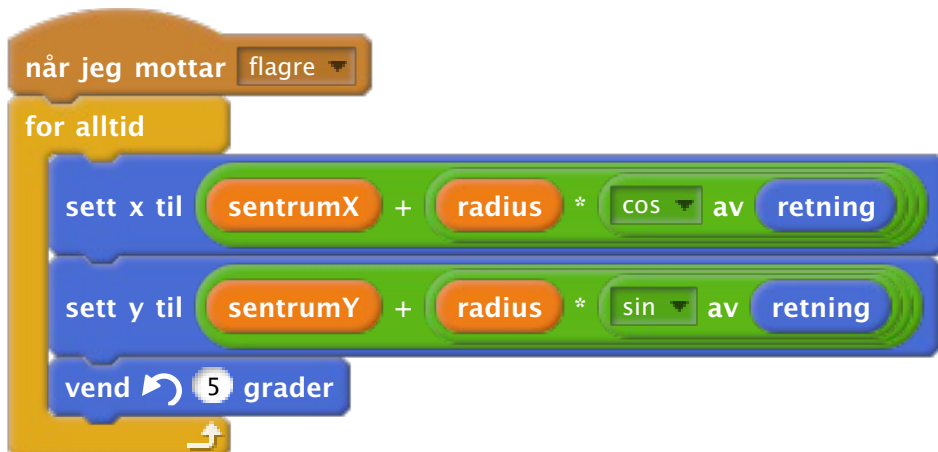
Vi skal nå se hvordan vi kan gjøre om denne kaotiske dansen til noe som ligner på et flagrende flagg.

### ✓ Sjekkliste

- ☐ I forrige steg klonet vi disker til tilfeldige steder. Nå skal vi legge dem ut litt mer organisert. Skriv om skriptet som legger ut diskene dine slik:



- ☐ La så **flagre** -skriptet være det samme som du brukte til å danse tidligere:



- ☐ Kjør programmet ditt. Du vil nå se et flagg bestående av mange røde disker som beveger seg i sirkel. Men nå kommer det morsomme: Vi kan la de røde diskene bevege seg litt i utakt!

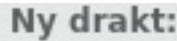
Legg til klossen  etter  -klossen, og prøv programmet ditt igjen. Hva har skjedd?


## Steg 5: Det norske flagget

Nå skal vi se hvordan vi kan tegne flagget i forskjellige farger.

### ✓ Sjekkliste

- ☐ Tegn først to nye drakter, begge kopier av den røde disken. Den første skal være en hvit disk mens den andre skal være en blå disk. Gi de tre diskfigurene dine navnene **r**, **h** og **b** slik at navnet er første bokstaven i fargen på disken: **r**ød, **h**vit og **b**lå.



- når  klikkes
- sett  til

rrrrrhbbhrrrr

- bytt drakt til bokstav nummer i flagg

for alltid

- flagg

```
rrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrr
hhhhhhbbhhhhhhbbbbbrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr
rrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrr
rrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrrrrrrrrhbbhrrrr
```





- mod

**bokstav nummer i flagg**

## Prøv selv

- 



- ☐ Du kan også leke litt med de forskjellige tallene for en litt annerledes animasjon. Prøv for eksempel med  i klossen etter  .
- ☐ Du kan endre utseendet på flagget underveis. For eksempel om du bruker en  -kloss kan du endre verdien av  basert på hvilke taster som trykkes. Fargene i flagget vil da også oppdatere seg.

**Lisens:** [CC BY-SA 4.0](#) **Forfatter** Sverre Oskar Konestabo og Geir Arne Hjelle