

□ Flagg



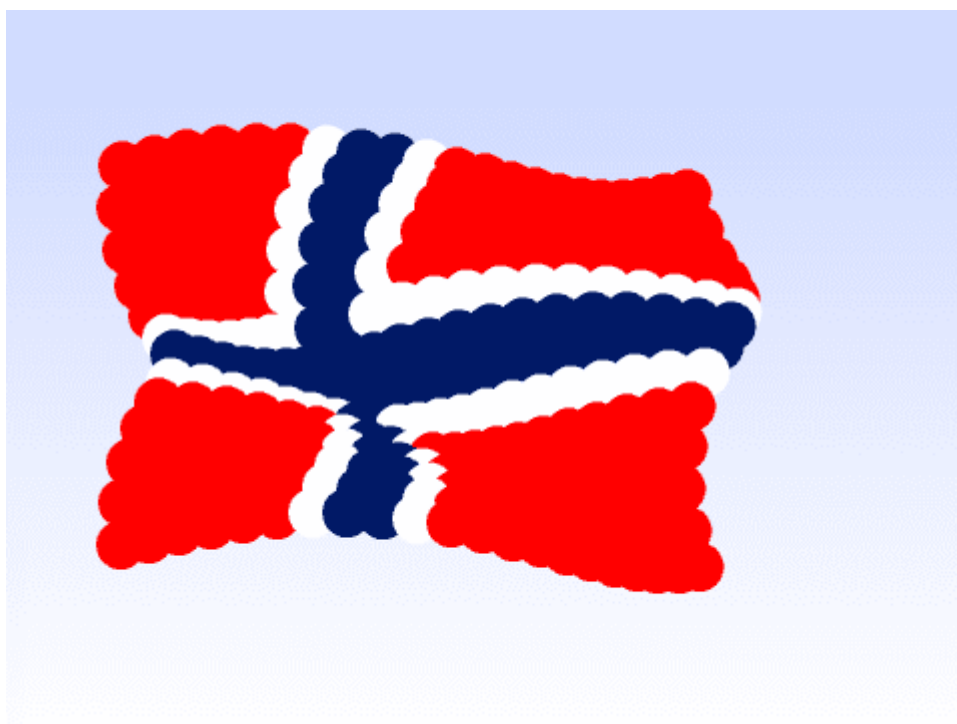
PÅ BOKMÅL



LAST NED PDF

Introduksjon

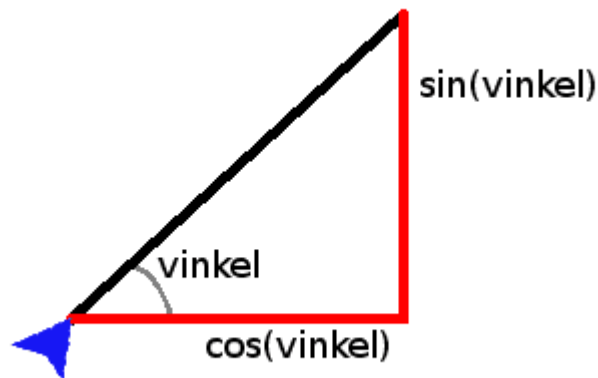
I denne oppgåva ser me nærare på korleis me kan lage mønstre og animasjonar ved hjelp av litt matematikk. Det me skal gjere er å teikne eit flagg som vaiar i vinden.



Steg 1: Gå i sirkel

Tidlegare har me sett fleire måtar me kan få figurar til å gå i sirkel på. Til dømes i [Soloball](#) der me flytta figuren vekk frå sentrum og brukte `peik mot [musepeikar v]` for å flytte figuren i sirkel.

Her skal me bruke to matematiske funksjonar, *sinus* og *cosinus* for å ha meir kontroll over korleis sirkelrørsla skjer. Du kjenner kanskje ikkje desse endå, men det er ikkje så farleg. Enkelt forklart hjelper dei oss å finne ut kor lange sidene i ein rettvingla trekant er i forhold til kvarandre. I figuren under fortel *cosinus*, `cos(vinkel)` oss kor lang den vassrette streken er i forhold til den skrå streken.



No skal me bruke dette til å teikne sirklar!



Sjekkliste

- ☐ Start eit nytt prosjekt.
- ☐ Skriv dette skriptet på kattefiguren:

```
når @greenFlag vert trykt på
avgrens rotering til [ikkje roter v]
for alltid
    set x til ((100) * ([cos v] av (retning)))
    set y til ((100) * ([sin v] av (retning)))
    snu @turnLeft (5) gradar
slutt
```



Test prosjektet

Klikk på det grønne flagget.

- ☐ Kva skjer? Forstår du kvifor kattefiguren flyttar seg rundt i ein sirkel?

Talet `100` fortel kor stor sirkelen skal vere (det er lengda av den skrå streken i figuren over). Prøv å endre talet (begge stadene) for å sjå kva som skjer.
- ☐ Kva skjer dersom du brukar ulike tal dei to stadene det står `100`?
- ☐ Korleis forandrar rørsla til katten seg dersom du forandrar talet `5` i `snu @turnLeft (5) gradar`?
Prøv òg kva som skjer om du brukar eit negativt tal!

Retning på sirkelrørsla

Har du lagt merke til at katten flyttar seg i motsatt retning av korleis pila i `snu @turnLeft (5) gradar`-klossen peikar? Det er fordi Scratch måler vinklar i motsatt retning av det som er vanleg i matematikken.

Steg 2: Flytt sirkelen

Til no har me berre teikna ein sirkel midt på skjermen. No skal me flytte den!



Sjekkliste

- ☐ Sidan me allereie flyttar figuren langs ein sirkel kan me ikkje bruke `gå til x: () y: ()`-klossar for å flytte heile sirkelen. I staden brukar me variablar. Lag to variablar som heiter `(sentrumX)` og `(sentrumY)` som berre gjeld *for denne figuren*.
- ☐ No kan me flytte sirkelen over ved å endre `(sentrumX)`- og `(sentrumY)`-variablane:

```
når @greenFlag vert trykt på
avgrens rotering til [ikkje roter v]
set [sentrumX v] til [-100]
set [sentrumY v] til [50]
for alltid
    set x til ((sentrumX) + ((100) * ([cos v] av (retning))))
    set y til ((sentrumY) + ((100) * ([sin v] av (retning))))
    snu @turnLeft (5) gradar
slutt
```



Test prosjektet

Klikk på det grøne flagget.

- ☐ Flyttar sirkelen seg til det nye sentrumet?

Prøv sjølv

- ☐ Legg til ein ny variabel `(radius)` som berre gjeld *for denne figuren*. Kan du bruke den til å styre kor stor sirkelen er? Det tyder at `radius` skal seie kor lang den skrå streken i figuren i starten av oppgåva skal vere.

Du treng ein `set [radius v] til []`-kloss i tillegg til å bruke `(radius)` to stader i koden din.

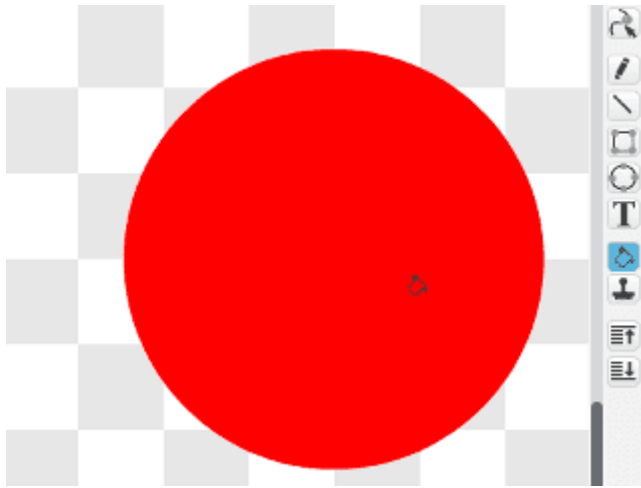
Steg 3: Dansande diskar

No skal me prøve å få mange figurar til å gå i sirkel samstundes.



Sjekkliste

- ☐ Teikne ei ny drakt på figuren din. Bruk vektorgrafikk og lag ein raud fylt sirkel (ein fylt sirkel kallast ein disk). Den kan vere ganske liten, til dømes `20 x 20` pikslar.



- ☐ Me kan lage mange raude diskar ved å klonе den me laga i førre punkt. Del koden din i to delar, og endre den slik:

```
når @greenFlag vert trykt på
gjenta (99) gongar
  set [sentrumX v] til (tilfeldig tal frå (-150) til (150))
  set [sentrumY v] til (tilfeldig tal frå (-100) til (100))
  set [radius v] til [50]
  lag klon av [meg v]
slutt

når eg startar som klon
for alltid
  set x til ((sentrumX) + ((radius) * ([cos v] av (retning))))
  set y til ((sentrumY) + ((radius) * ([sin v] av (retning))))
  snu @turnLeft (5) gradar
slutt
```

- ☐ Om du køyrer programmet dinn no skal du sjå 99 diskar som "dansar" og ein som står i ro på skjermen. Det ser kanskje litt kaotisk ut!

Kaaset kjem av at Scratch brukar litt tid på å starte kvar klon. Me kan få orden på det ved å få dei til å starte samstundes:

Bytt når eg startar som klon med når eg får meldinga [dans v], og legg til ein send meldinga [dans v]-kloss etter gjenta (99) gongar-løkka.

Om du køyrer programmet ditt att ser du at alle diskane "dansar" i takt. Kva likar du best?

Steg 4: Flagrande flagg

No skal me få denne kaotiske dansen til å bli noko som liknar på eit flagrande flagg.



Sjekkliste

- ☐ I førre steg klona me diskar til tilfeldige stader. No skal me leggje dei ut på ein bestemt måte for å etterlikne eit flagg. Skriv om skriptet som legg ut diskane slik:

```
når @greenFlag vert trykt på
vis
```

```

set [radius v] til [25]
set [sentrumX v] til [-160]
gjenta (19) gongar
  set [sentrumY v] til [-100]
  gjenta (14) gongar
    lag klon av [meg v]
    endra [sentrumY v] med (16)
  slutt
  endra [sentrumX v] med (16)
slutt
send meldinga [flagre v]
gøym

```

- ☐ No kan `flagre`-skriptet vere det same du brukte til å lage "dansen" tidlegare.

```

når eg får meldinga [flagre v]
for alltid
  set x til ((sentrumX) + ((radius) * ([cos v] av (retning))))
  set y til ((sentrumY) + ((radius) * ([sin v] av (retning))))
  snu @turnLeft (5) gradar
slutt

```

- ☐ Kjør programmet ditt. No skal du sjå eit flagg som består av mange raude diskar som består av mange raude diskar som beveger seg i sirkel. Men no kan me la dei raude diskane bevege seg litt i utakt!

Legg til klossen `snu @turnRight (1) gradar` **etter** `lag klon av [meg v]`-klossen, og prøv programmet ditt att. Kva har skjedd?

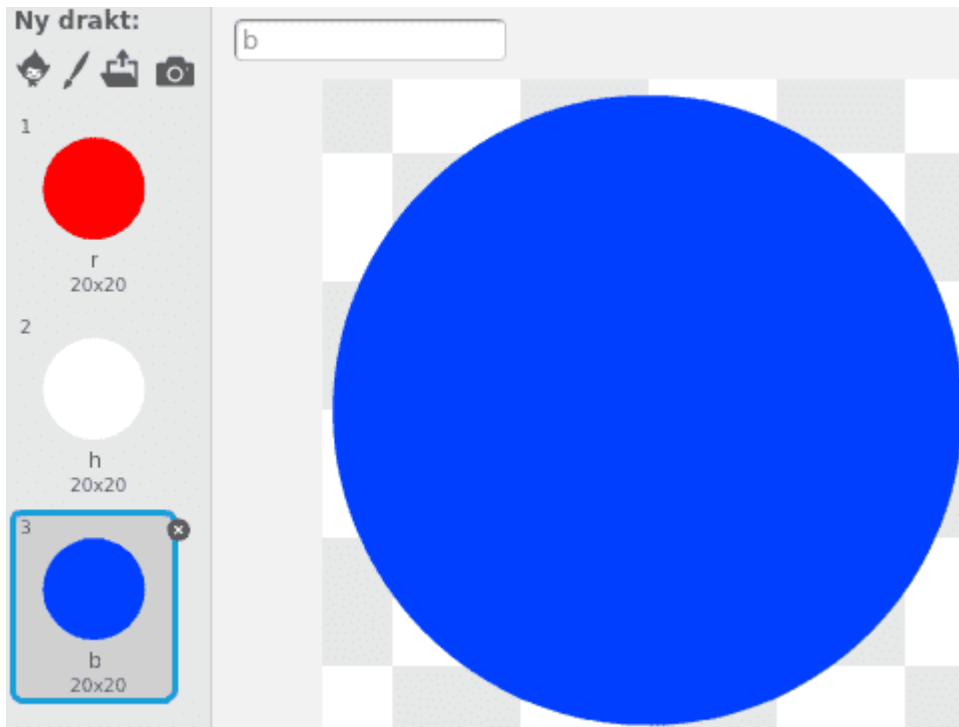
Steg 5: Det norske flagget

No skal me teikne flagget med ulike fargar.



Sjekkliste

- ☐ Teikne to nye drakter, begge som kopiar av den raude disken. Den fyrste skal vere kvit og den andre skal vere blå. Gi figurane namna `r`, `k` og `b` slik at namnet er den fyrste bokstaven i farga på disken: `raud`, `kvit` og `blå`.



- ☐ No skal me bruke ein ny variabel (`flagg`) for å beskrive fargane i flagget. Lag variabelen for alle figurar og legg til koden

```
når @greenFlag vert trykt på
set [flagg v] til [rrrrrkbbkrrrrr]
```

Bokstavane rrrrrkbbkrrrrr beskriv at me vil ha fem raude diskar, ein kvit, to blå, ein kvit og fem raude diskar.

- ☐ For å kunne bruke den nye `(flagg)`-variabelen må kvar disk vite kva nummer den har. Lag ein ny variabel `(nummer)` som berre gjeld *for denne figuren*.
- ☐ Legg til `set [nummer v] til [1]` rett under `vis` og endra `[nummer v]` med `(1)` rett under `lag` klon av `[meg v]`-klossen.
- ☐ Til slutt skal me endre drakt på diskane. Legg til

```
byt drakt til (bokstav (nummer) i (flagg))
```

øvst i for alltid-løkka i flagre-skriptet.

- ☐ Kjør programmet ditt. Den venstre delen av flagget ditt skal ha fått fargane til det norske flagget. For å farge heilt flagget må me gi mange fleire bokstavar til `(flagg)`-variabelen. Heile det norske flagget har bokstavane

[illegible]

- Ein enklare måte å teikne flagg på er å seie at me vil at mønsteret skal gjenta seg sjølv. Då treng me ikkje å skrive ein bokstav for kvar disk. Til det kan me bruke $() \bmod ()$ -klossen.

Bytt (bokstav (nummer) i (flagg)) **med** (bokstav ((nummer) mod (lengda til (flagg))) i (flagg)) i byt drakt til [v]-klossen. Om du prøver programmet att ser du at flaggmønsteret blir gjenteke. Prøv å sette (flagg) til rkb for å sjå det.

Prøv sjølv

- ☐ Teikne dine egne flagg. Om du treng flere fargar kan du berre lage flere drakter. Pass på at kvar drakt har ein bokstav eller eit tal som namn.
- ☐ Du kan leike litt med tala for å endre på animasjonen. Prøv med `snu @turnRight (25) gradar i klossen etter lag klon av [meg v]`.
- ☐ Du kan endre korleis flagget ser ut undervegs. Til dømes kan du bruke ein `når [v] vert trykt-kloss` for å endre verdien av `(flagg)` basert på kva tastar som blir trykka. Då vil fargene i flagget oppdatere seg.

Lisens: CC BY-SA 4.0